中国大学MOOC

DIY智慧小屋 带你玩转物联网

DIY智慧小屋 带你玩转物联网

课程配套操作手册



（课程二维码）

硬件搭建、软件编写、系统调试全覆盖

浙江大学 ZHEJIANG UNIVERSITY

阿里云

DIY 智慧小屋操作手册

V1.0

「手册说明」

“DIY 智慧小屋带你玩转物联网”是浙江大学教授团队联合阿里云 MVP（最具价值专家）在中国大学 MOOC 平台上推出的物联网全栈理论+实践的一门课程，在推出后获众多同学们的关注，本《DIY 智慧小屋操作手册》为课程配套的软硬件操作手册，旨在配套在线课程，记录实践中的各个操作步骤，软硬件开发技巧和硬件获取渠道，帮助同学们轻松完成智慧小屋的搭建和运行，在实践中取得收获。

此操作手册以操作为主，建议结合 MOOC 视频了解理论知识以及系统详解，在学习完章节视频后，再阅读此操作手册进行实践。



图：课程二维码

课程链接：http://www.icourse163.org/course/ZJU-1206632831

本操作手册分为三个部分，**课程综述**、**嵌入式篇**和**平台篇**。

课程综述部分将着重介绍课程使用到的嵌入式设备的系统组成、硬件规格、硬件设备接线，以及安装时的一些注意事项，可以提前进行材料准备。

嵌入式篇则侧重介绍 Arduino 嵌入式平台，嵌入式开发所依赖的一些库函数、IDE 工具以及嵌入式程序烧写到系统调试的所有操作步骤，嵌入式篇对应在线课程的第三、四、五、六章内容，建议先完成各章节课程学习测试，再进实践操作。

平台篇则侧重介绍设备数据上云后使用物联网平台提供的 IoT Studio 进行应用程序快速开发，包含平台中设备导入/调试、服务构建、Web 搭建和 App 搭建等，平台篇对应在线课程的第七、八、九章内容，同样建议大家先完成在线课程学习和测试，再依照手册进行实践。

目录

一、 课程综述

1.1 课程介绍

智能家居一直是物联网关注的热点之一，是物联网重要落地平台，智能家居的实现将极大地便利人们的日常生活，有着非常重要的现实意义。在课程中，我们通过将物联网的应用落实到智能小屋架构上，可以直观地看到数据的采集、数据的迁移、云端处理数据以及整个控制逻辑的构架,课程囊括了设备端数据的采集、数据上云、云端处理、基于物联网平台的 WEB 页面控制和显示以及构建手机 APP 控制的整个完整流程。

为了配合课程的教学，我们搭建了一个物联网智慧小屋的模型，其中包括了一般的家居所需的装置，包含的传感器件有以下的部件：室内温湿度传感器、室外温湿度传感器、PM2.5 传感器（使用 CO2 传感器替代）、可燃气体传感器、土壤湿度传感器、光敏传感器，包含的执行模块有：空调、换气扇、水泵、可调光以及调色的 LED 灯。传感信号是输入信号，由 Arduino 模块进行采集，数据采集后，由无线模块进行传输，在小屋实例中，我们采用了 WiFi 通过 AP 进行传输的方式，通过公网传送到云平台端，实现数据的转发、处理等。云端将控制信号发回给设备端执行。通过智慧小屋功能完整的演示，我们将传感器和执行机构之间的逻辑关系关联起来。目前设定的五个场景是：

1、室内的温度与空调关联，温度高了，开启空调制冷。平时我们开空调都是温度高了，自己用遥控器开启空调。在智慧小屋里，只需要给定相应的参数，就可以按照设定的参数，空调就可以自主运行，开启空调降温和去湿。

2、当可燃气体传感器超标时，风扇、蜂鸣器动作

3、光敏电阻感知外界的光线的强弱，用以判定夜幕降临，在光线暗淡时，拉上窗帘、开启室内的灯光

4、PM2.5 传感器感知数据超标时，将开启风扇（模拟空气净化器）换气。

5、检测植物的土壤的湿度，在偏低时启动水泵，进行浇灌作业。

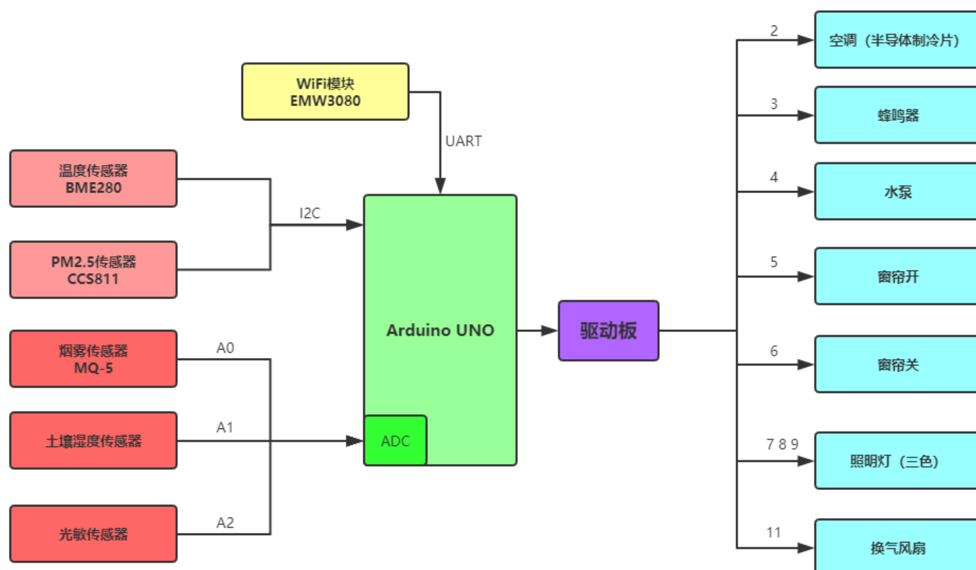
物联网大致可以分成：感知层、网络层、平台层和应用层。在课程中，我们可以直观看到物联网的分层架构，其与智慧小屋里相对应的关系为：感知层主要是负责通过传感器设备来识别和收集信息，我们搭建了通过 Arduino 采集不同接口形式的传感数据的实例来学习感知层的构架； 网络层负责安全的把这些信息进行传输，我们通过 AT 指令，将数据通过 WiFi 传送到物联网平台端。平台层负责数据的鉴权、接入和转发，我们以阿里云为例，讲述了接入云平台需要的鉴权方式、MQTT 协议。应用层负责结合具体的应用需求，利用 IoT Studio 这些先进的可视化工具，将建立服务编排，详细讲解了如何创立直观的 Web 显示、调度和控制的页面，并可方便地编制出手机控制的 APP，通过实例让大家能快速掌握构架物联网系统原型的能力，并在云平台上对数据进行计算、处理、挖掘，来实现智能化的物联网应用。

1.2 材料清单

见手册最后部分附件 1。

1.3 硬件组成

智慧小屋基于 Arduino 嵌入式开发平台的系统构成如下图所示：



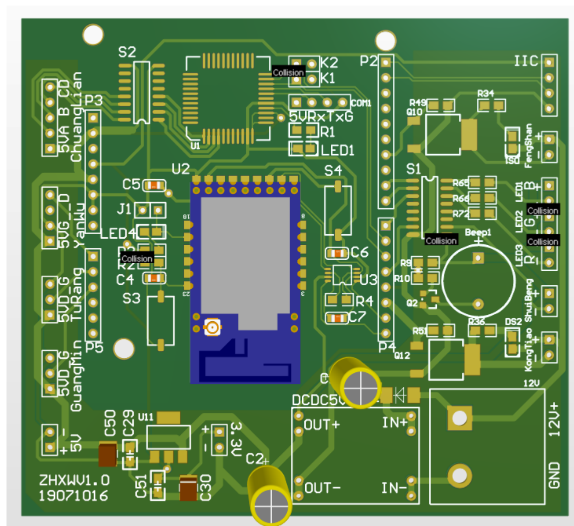
图：智慧小屋的硬件系统构成

如图所示为智慧小屋的硬件架构图，大致可以分为四个部分，左侧红色的为传感器，右侧蓝色的为执行器，上方黄色的 WiFi 通讯模块和中间绿色的 Arduino UNO 开发板。

右侧执行器中的水泵、空调、照明灯和换气风扇属于功率器件，需要大电流的驱动，Arduino 开发板显然无法提供这么高的功率。驱动板的设计与加入就是为了解决这个问题。

1.4 硬件接线

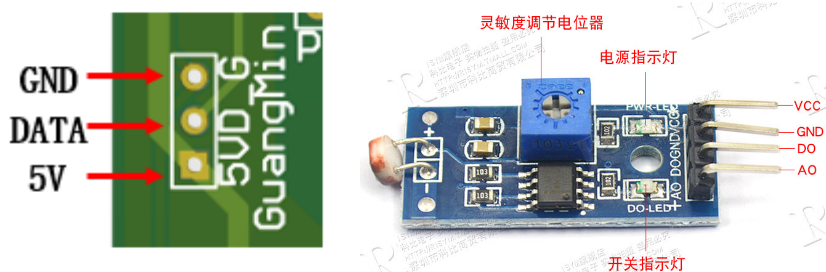
硬件接线示意图以及注意事项



图：小屋 PCB 设计三维预览图

上图所示为小屋的 PCB 设计三维预览图。中间蓝色的部分为集成了天线的 WiFi 模块，WiFi 模块两侧丝印 P2、P3、P4、P5 为适配了 Arduino UNO 开发板和 Arduino Mega 2560 开发板的排针。驱动板周围的一圈为驱动板对外的各种接口。下面从左下方开始，沿顺时针方向，分别对每个端口的接线进行说明。

① 光敏传感器端口

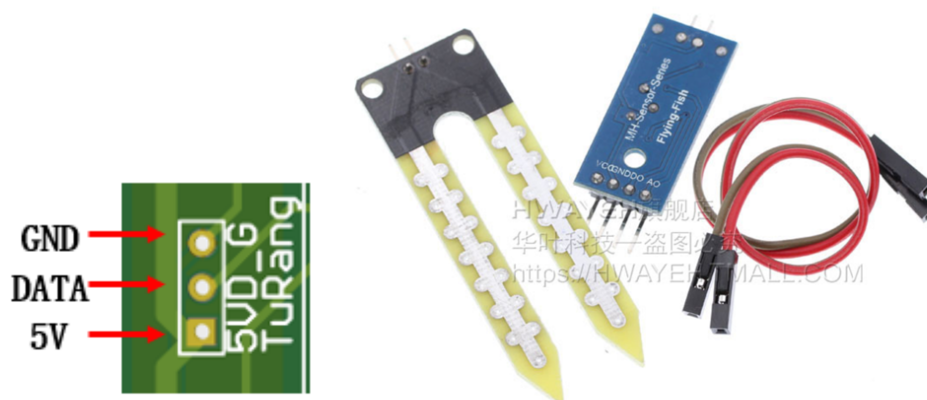


图：光敏传感器

上图左为驱动板的光敏传感器接口，位于驱动板的左下角。为了方便说明接线，右侧为常见的 4 线光敏传感器引脚示意图。

- 驱动板接口的 GND 为地线，接右侧光敏传感器的 GND。
- 驱动板接口的 DATA 为模拟信号线，接右侧光敏传感器的 AO
- 驱动板接口的 5V 为供电电源线，接右侧光敏传感器的 VCC

② 土壤传感器端口

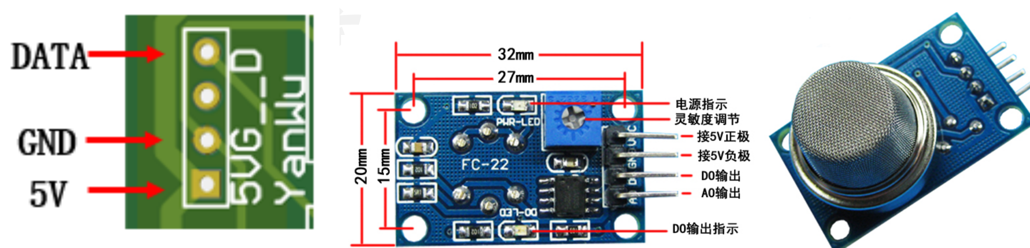


图：土壤传感器

上图左为驱动板的土壤湿度传感器接口，位于光敏传感器接口的上方。同理，为了方便说明，右侧选择了常见的四线土壤湿度传感器模块，从左至右端口丝印依次为 VCC、GND、DO、AO。

- 驱动板接口的 GND 为地线，接右侧土壤湿度传感器的 GND。
- 驱动板接口的 DATA 为模拟信号线，接右侧土壤湿度传感器的 AO
- 驱动板接口的 5V 为供电电源线，接右侧土壤湿度传感器的 VCC

③ 可燃气检测传感器端口



图：可燃气传感器

上图左为可燃气检测传感器接口，丝印标注为 YanWu，位于土壤湿度传感器接口的上方，DATA 与 GND 之间的端口为悬空，不需要接。图右为常见的可燃气检测传感器接线图与正面示意图。

- 驱动板接口的 GND 为地线，接右侧可燃气检测传感器的 GND。示意图中 GND 标注为接 5V 负极实际是为了方便电压比较器工作，我们不需要数字量输出，接 GND 就好。
- 驱动板接口的 DATA 为模拟信号线，接右侧可燃气检测传感器的 AO
- 驱动板接口的 5V 为供电电源线，接右侧可燃气检测传感器的 VCC

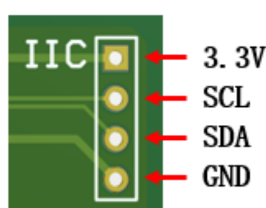
④ 窗帘步进电机端口



图：窗帘步进电机 PCB 端口

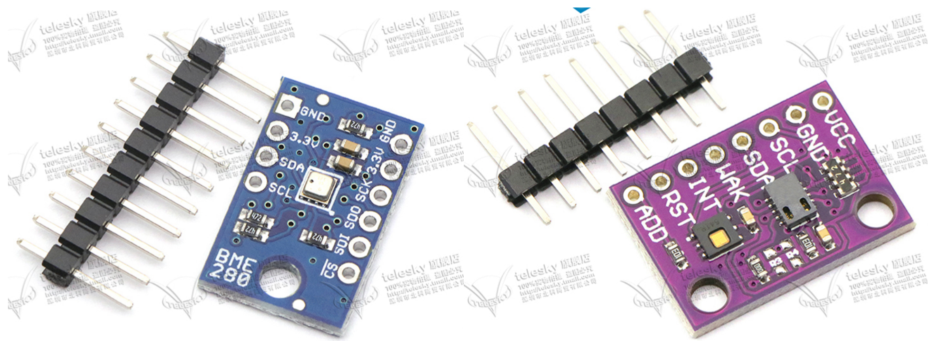
上图左为窗帘步进电机端口，位于驱动板左上角。图右为常见的五线四相减速步进电机。步进电机的连接端使用了 XH-2.54 插接端口，具有防呆设计，若方向错误无法插入。

⑤ IIC 传感器端口



图：IIC 传感器 PCB 端口

上图左为 IIC 传感器端口，位于驱动板右上角。IIC 为总线协议，支持并联接入多个从设备。小屋中使用的温湿度传感器 BME280 和空气质量传感器 CCS811 就是并联接到这个端口上的。3.3V 为模块供电端口。



图：BME280 环境传感器和 CCS811 环境传感器

- 上图左为常见的 BME280 模块，3.3V、SDA、SCL、GND 对应丝印接好即可。
- 上图右为常见的 CCS811 模块，除了端口的四根线外，RST 引脚也要接到 GND。

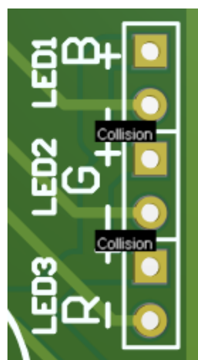
⑥ 风扇端口



图：PCB 风扇端口

上图左为驱动板风扇端口，位于 IIC 传感器端口的下方。图右为常见的 12V 风扇，可以看到采用了 XH-2.54 插接件，插入连接即可。

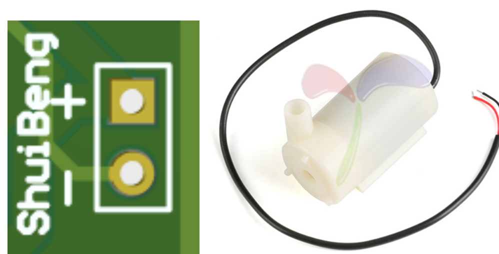
⑦ LED 三色灯端口



图：LED 三色灯 PCB 端口

上图为驱动板 LED 三色灯端口，位于风扇端口的下方，三种颜色的 LED 各有两个端口，正极和负极，从上到下依次为，蓝色，绿色，红色。实际在小屋功能中，LED 灯的颜色并不加以区分，接对正负极即可。

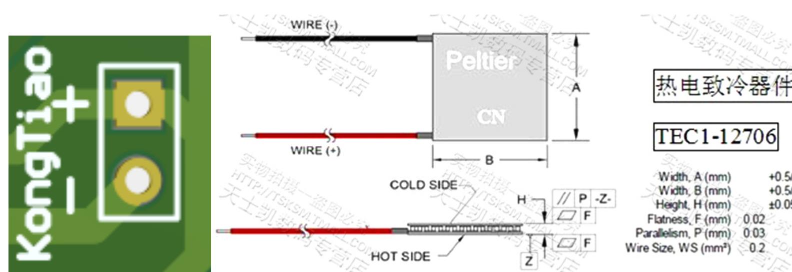
⑧ 水泵端口



图：水泵 PCB 端口

上图左为驱动板水泵端口，位于 LED 三色灯端口下方。图右为水泵，采用直流电机，红色线为正极，黑色线为负极，对应端口的正负极丝印标注进行连接。

⑨ 空调端口



图：空调（半导体制冷片）PCB 端口

上图左为驱动板空调端口，位于水泵端口下方。图右为半导体制冷片的示意图，同样有红黑两根线，红色代表正极，黑色代表负极。对应端口的正负极丝印标注连接。

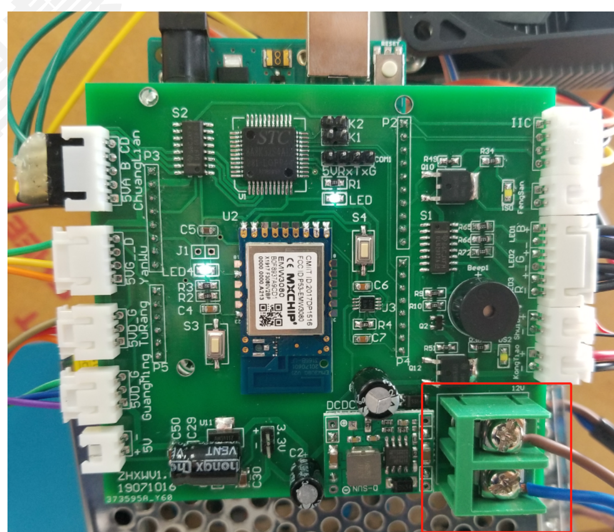
1.5 注意事项

驱动板使用 12V 电源提供供电，供电电源最大电流不低于 3A。



图：电源模块

上图所示为一种适用的电源，电源输入为市电，必须使用三头插座电线，正确连接零线、火线和地线，确保电源外壳接地，以免发生意外。



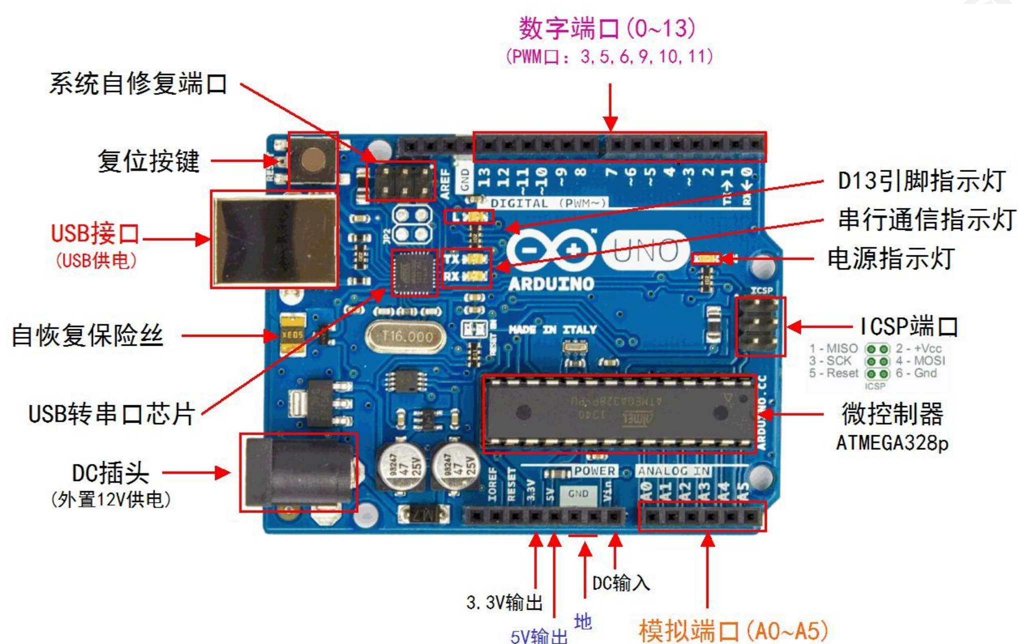
图：驱动板供电端口

上图所示为正确安装驱动板的示例，Arduino UNO 开发板通过排插连接到驱动板后无需额外供电，右下角红框内为驱动板供电端口，12V 与 GND 务必正确连接。

二、嵌入式篇

2.1 嵌入式综述

Arduino UNO 使用爱特梅尔公司（ATMEL）设计生产的 ATmega328P 作为微控制器。板载资源如下图所示：



图：Arduino UNO 开发板

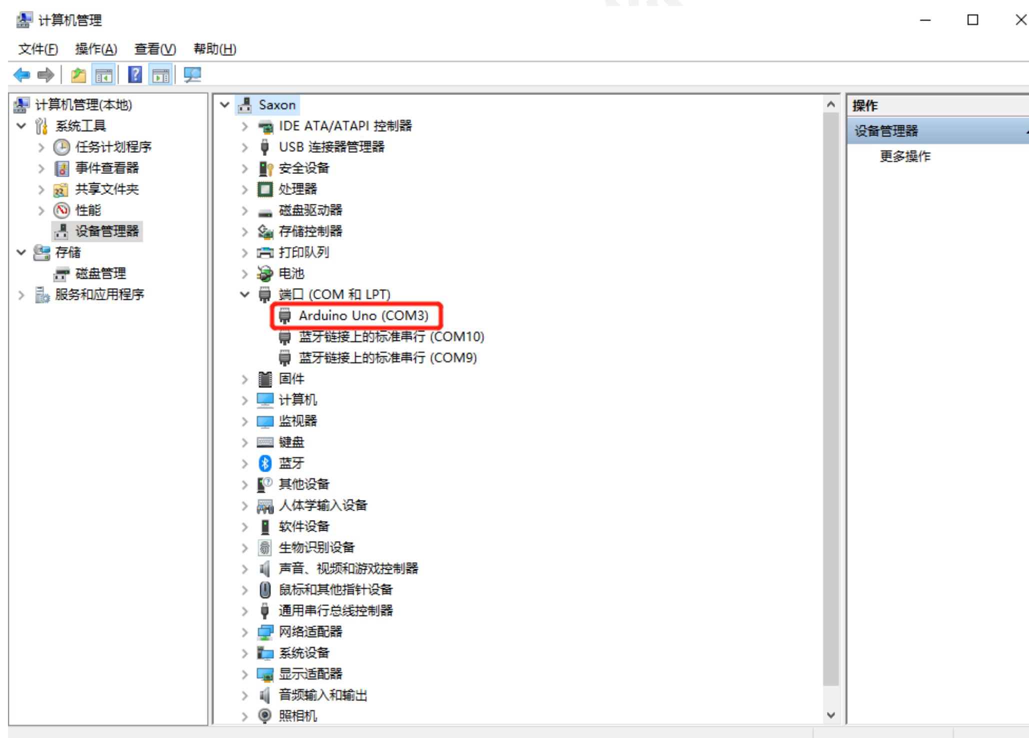
开发板可以通过 USB-Type-B 进行供电，也可以通过 DC 插头进行供电。若使用 DC 插头进行供电，请务必确认电源接口的正负极为内正外负，电压在 7~12V 之间均可，越接近 7V 越好。教程中无特别说明，默认通过 USB 接口将开发板与电脑连接，既满足了供电需求也保证了数字通信。连接后如下图所示。



图：开发板连接示意图

可以看到标有“ON”字样的电源指示灯常亮，说明连接成功。

接着打开 Windows 系统的设备管理器，查看数据连接情况，如图所示可以看到连接的开发板被正确识别，并注明使用了 COM3 端口。



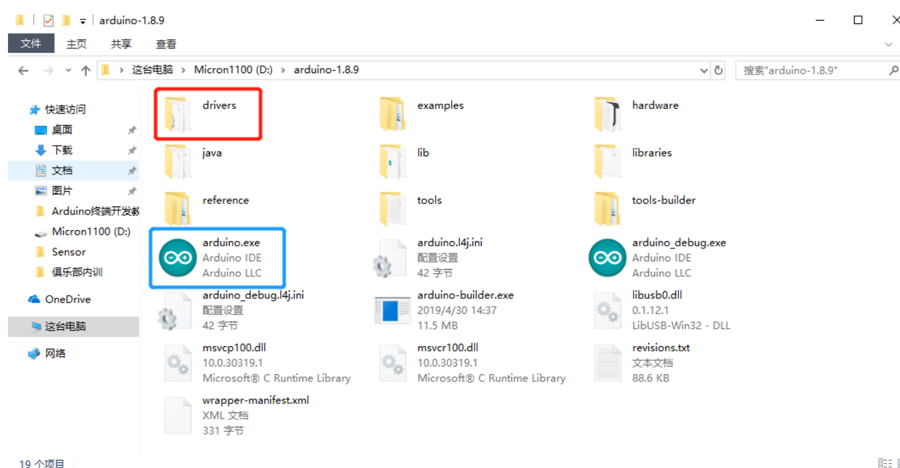
图：设备管理器中查看 Arduino Uno 连接

若您的设备管理器显示未识别的设备，不必惊慌，接下来的 IDE 安装会解决这个问题。

2.2 环境搭建

➢ STEP1 下载 IDE

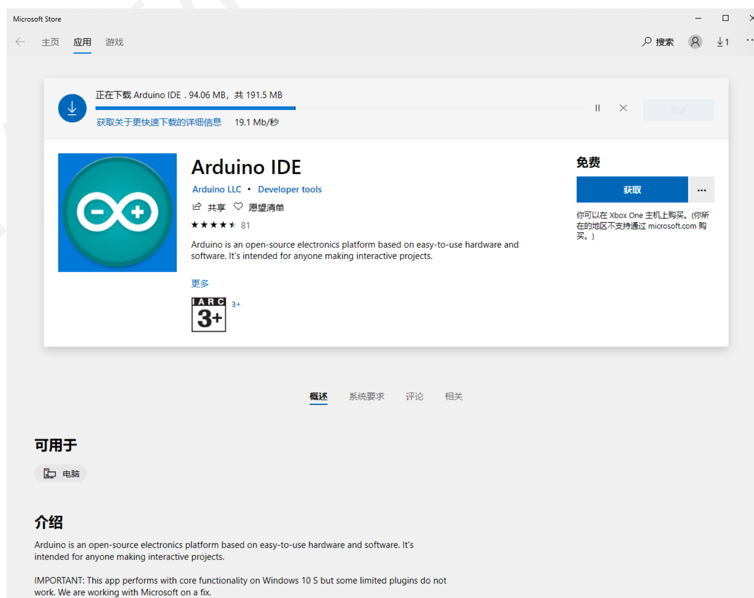
访问https://www.arduino.cc/en/Main/Software，下载1.8.9的免安装包，直接解压文件。



图：解压后的文件夹

如上图所示，解压后的文件中包含 drivers 文件夹，前文提到，若您的电脑没有正确的识别开发板，可以在提示中指定驱动搜索目录到 drivers 文件夹，即可解决问题。

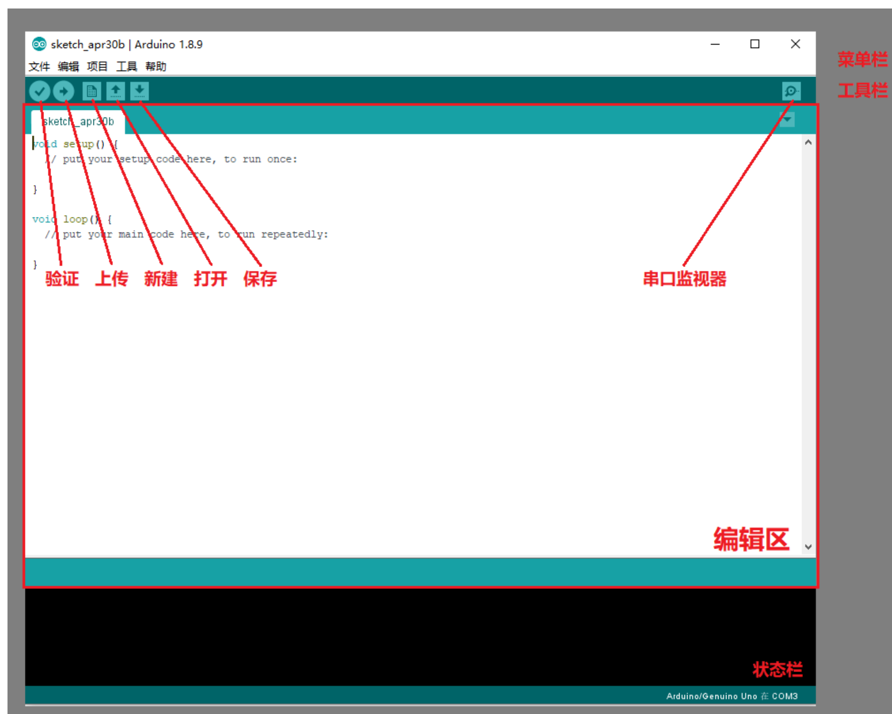
Arduino IDE 已上架 Microsoft Store，若您使用 Windows 8.1 或 10 版本操作系统，直接在应用商店搜索“Arduino IDE”，单击**获取**即可下载安装。



图：在 Windows10 应用商店下载 Arduino IDE

➢ STEP2 运行 Arduino IDE

运行文件夹中的 arduino.exe 打开编译器如下图所示。

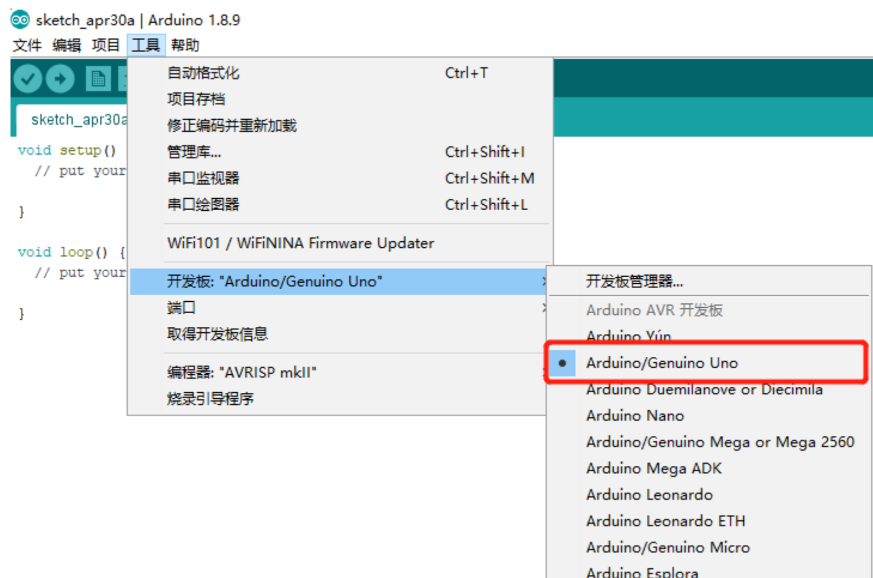


图：Arduino IDE 界面

Arduino IDE 如上图所示可分为四个部分：顶部为菜单栏和工具栏，其中工具栏有 6 个快捷按钮，功能如图；中间为编辑区；底部为状态栏，用于输出编译或调试信息。

➢ STEP3 选择开发板型号

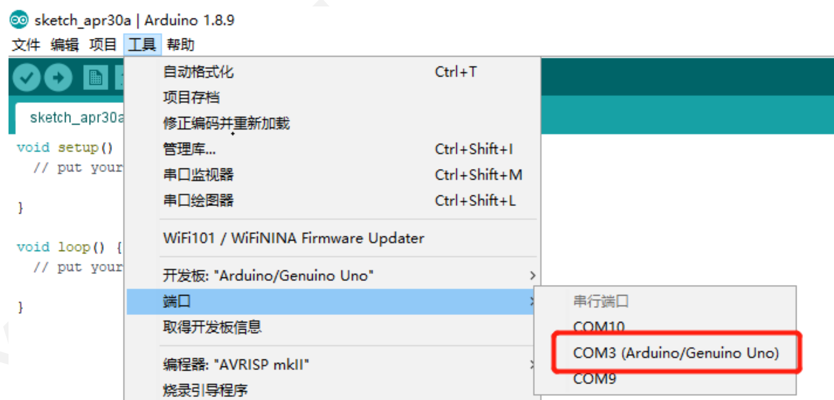
在工具菜单下开发板中选择 Arduino/Genuino Uno。



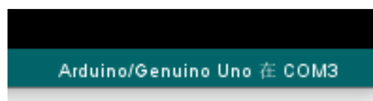
图：开发板型号选择

➢ STEP4 选择端口

接着在**端口**中选择开发板使用的端口，如前文设备管理器中所示，本实验中连接到了 COM3 端口，根据提示选择对应的端口。

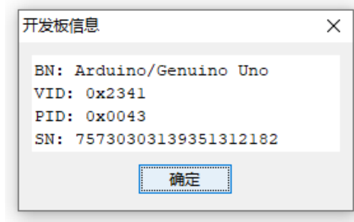


图：端口选择

更改成功后 IDE 的右下角提示信息变为 

。

接着单击**工具**菜单下的**取得开发板信息**，弹出下图所示对话框，说明连接成功。



图：开发板连接成功信息

2.3 软件烧写

➢ STEP1 下载资料包

https://pan.baidu.com/s/10dgDeJu5bs-ktUyhvaVlkA

访问以上链接下载安装包。

名称	大小	压缩后大小	类型
..			文件夹
Demo			文件夹
Adafruit_BME280_Library-master.zip	455,214	453,905	ZIP 文件
Adafruit_CCS811-master.zip	9,842	8,325	ZIP 文件
Adafruit_Sensor-master.zip	7,830	7,366	ZIP 文件

图:压缩包内容

如上图所示，下载后的压缩包内共有一个文件夹和三个.zip 文件。demo 文件夹内为要烧写到 Arduino UNO 开发板中的代码。其余三个.zip 压缩文件为需要添加到 IDE 中的库文件。

➢ STEP2 添加库文件

运行 Arduino IDE，在项目菜单下找到“添加.ZIP 库...”如下图所示：

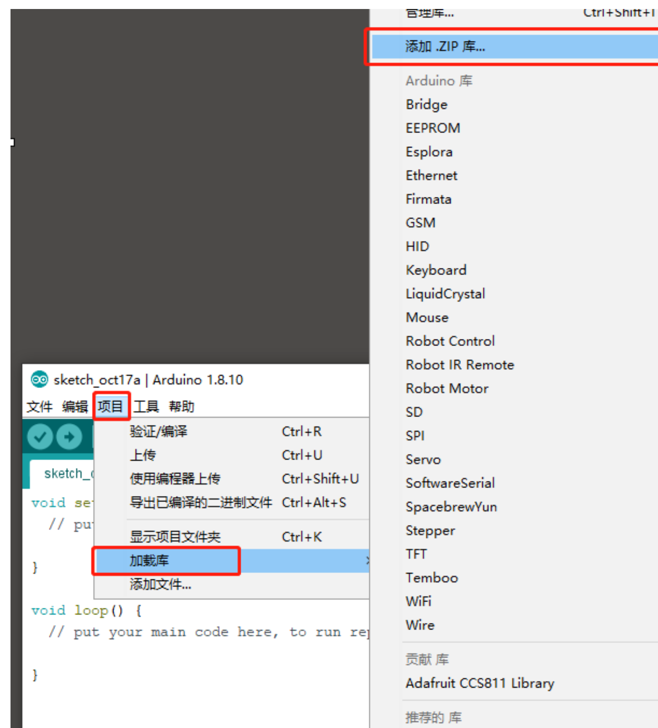


图:添加.ZIP 库...

在弹出的窗口中找到解压出的三个.zip 文件，如下图所示：

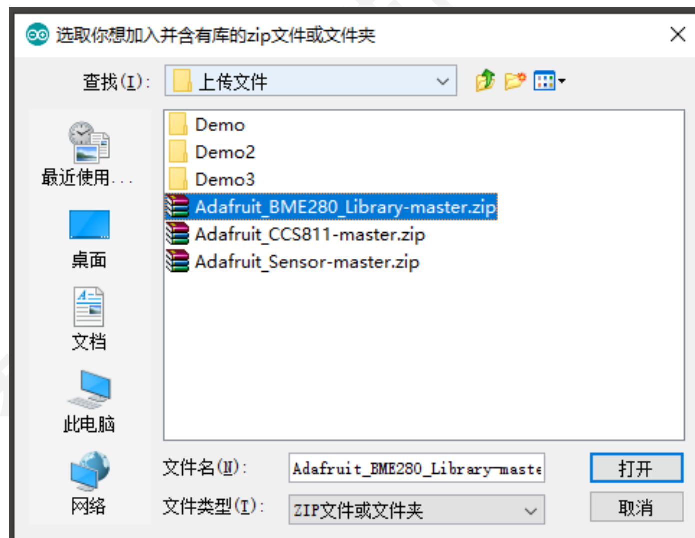
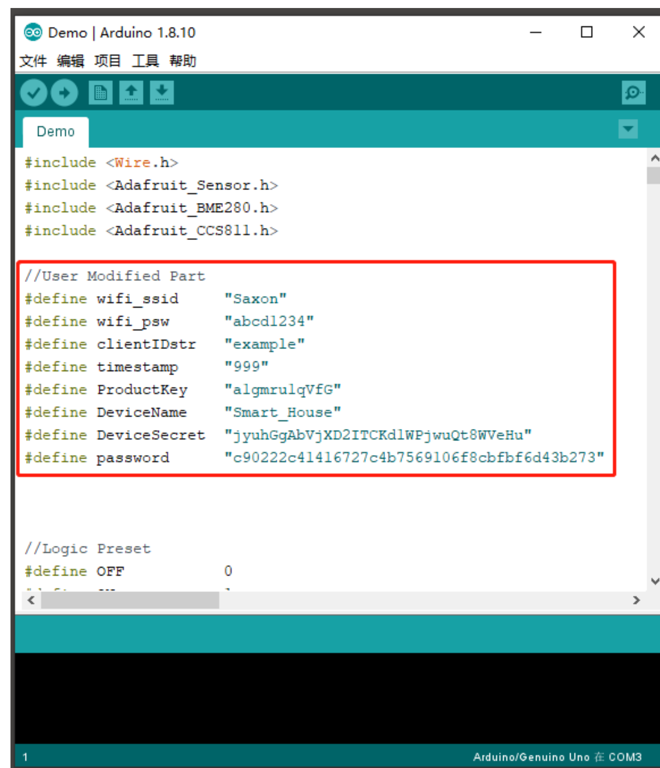


图:解压的.zip 文件

单击打开添加，同样的操作，添加另外两个.zip 文件。

➢ STEP3 修改代码

打开 demo 文件夹中的 Demo.ino，使用三元组等信息替换内容。



图：修改部分

如上图所示，此处我们先修改 wifi_ssid 和 wifi_psw 为附近可用 WiFi 热点的名称和密码。其余信息的获取与替换，在平台部分会有详细讲解。

➢ STEP4 上传代码

参照平台的部分，获取所有需要替换的字段，代码修改好后，单击上传按钮烧写开发板。

2.4 运行测试

运行测试分为两部分，嵌入式本地运行测试以及嵌入式和云平台的联调测试。

■ **嵌入式运行测试：**

将烧写好的开发板通过排插连接到驱动板。开启电源供电。

蜂鸣器会提示当前程序的运行状态：

- 响一声代表 IIC 传感器初始化完成，若没有响请检查 IIC 传感器连接。
- 响两声代表 wifi 连接完成，若没有响请检查 wifi 配置信息。
- 响三声代表与阿里云物联网平台的连接已经建立。若没有响请确认平台篇部分替换的字段是否正确。

总计六次蜂鸣器的响声过后，小屋开始正常运行各种功能。除可燃气检测联动场景的消除静音需要正确接收下发的消息，其余场景的联动均在本地运行。大家可以根据场景进行各种测试。检查各类传感器，执行器的状况。

■ **平台联调测试**

可参考平台篇内容开通阿里云账号。

➢ STEP1 创建产品

https://iot.console.aliyun.com/lk/summary

访问以上链接登录阿里云物联网平台，若没有账户，可使用支付宝扫码快速注册。



服务器选择华东 2，左侧的设备管理中找到产品。单击进入产品管理界面。在产品管理界面的右上角找到蓝色的创建产品，新建产品如图。

单击完成后等待几秒，在产品列表中将看到新创建的智慧小屋，单击查看进入产品详情界面。

产品详情页面上方找到功能定义，单击进入功能定义选项卡。

从自定义功能栏目右边找到蓝色的添加功能。添加功能如下图。

添加自定义功能

* 功能类型:
属性 服务 事件

* 功能名称:
室内温度

* 标识符:
Temperature

* 数据类型:
float (单精度浮点型)

* 取值范围:
-100 ~ 100

* 步长:
0.01

单位:
摄氏度 / °C

* 读写类型:
读写 只读

描述:
请输入描述
0/100

确认 取消

重复同样的操作，继续添加其他属性，注意属性的标识符与代码中属性的命名一致，蜂鸣器为读写类型。

自定义功能 添加功能

功能类型	功能名称	标识符	数据类型	数据定义	操作
属性	室内温度	RoomTemp	float (单精度浮点型)	取值范围:-50 ~ 100	编辑 删除
属性	空调状态	AC	bool (布尔型)	布尔值: 0 - 关 1 - 开	编辑 删除
属性	土壤湿度传感器读数	SoilHumi	int32 (整数型)	取值范围:0 ~ 1023	编辑 删除
属性	水泵状态	Pump	bool (布尔型)	布尔值: 0 - 关 1 - 开	编辑 删除
属性	可燃气传感器读数	GasDetector	int32 (整数型)	取值范围:0 ~ 1023	编辑 删除
属性	风扇状态	Fan	bool (布尔型)	布尔值: 0 - 关 1 - 开	编辑 删除
属性	蜂鸣器状态	Buzzer	enum (枚举型)	枚举值: 0 - 关闭 1 - 开启 2 - 静音	编辑 删除
属性	光敏传感器读数	LightDetector	int32 (整数型)	取值范围:0 ~ 1023	编辑 删除
属性	灯光状态	Light	bool (布尔型)	布尔值: 0 - 关 1 - 开	编辑 删除
属性	窗帘状态	Curtain	bool (布尔型)	布尔值: 0 - 关 1 - 开	编辑 删除
属性	类二氧化碳浓度	eCO2	int32 (整数型)	取值范围:400 ~ 8192	编辑 删除
属性	有机气体浓度	TVOC	int32 (整数型)	取值范围:0 ~ 1187	编辑 删除

全部添加完成后的属性如上图所示。

➢ STEP2 添加设备

在左侧导航栏中找到设备，单击进入设备管理页面，单击新建设备。添加设备如图。

单击确认后，会弹出设备的三元组信息窗口，如下图所示，单击一键复制。

新建记事本，将复制的三元组粘贴到记事本中。



➢ STEP3 计算哈希值

复制以下文字，粘贴到记事本中。

clientIDstr   example

timestamp 999

clientIdAAAdeviceNameBBBproductKeyCCCtimestampDDD



利用已有字段对红框内最后一行的信息进行替换：

- AAA 替换为 clientIDstr 后面的内容
- BBB 替换为三元组的 DeviceName
- CCC 替换为三元组的 ProductKey
- DDD 替换为 timestamp 后面的内容

替换后的效果如下图所示。

```
{
  "ProductKey": "a1gmru1qVfG",
  "DeviceName": "Smart_House",
  "DeviceSecret": "ykPpoww49PxVXDn0imY4tBfWxFIxE1Z8"
}

clientIDstr     example
timestamp       999
clientIdexampledeviceNameSmart_HouseproductKeya1gmru1qVfGtimestamp999
```

http://iot-face.oss-cn-shanghai.aliyuncs.com/tools.htm

访问链接打开签名算法计算工具：

- 将最后一行的内容填写到明文中
- 将 DeviceSecret 填写到秘钥中

单击 hmac-sha1 按钮生成哈希值如下图。

HmacMD5,HmacSHA1,HmacSHA256 签名算法验证

明文：clientIdexampledeviceNameSmart_HouseproductKeya1gmru1qVfGtimestamp999

秘钥：ykPpoww49PxVXDn0imY4tBfWxFIxE1Z8

hmac-md5 hmac-sha1 hmac-sha256

哈希值：d8304387381ed3acb633a673573951a0f170209f

复制生成的哈希值到记事本中，作为 password 字段替换代码内容。

➢ STEP4 修改代码

打开 Demo 样例代码，找到用户自定义部分如图。

```
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BME280.h>
#include <Adafruit_CCS811.h>

//User Modified Part
#define wifi_ssid      "Saxon"
#define wifi_psw       "abcd1234"
#define clientIDstr    "example"
#define timestamp      "999"
#define ProductKey     "algmrulqVfG"
#define DeviceName     "Smart_House"
#define DeviceSecret   "jyuhGgAbVjXD2ITCKdlWPjwuQt8WVeHu"
#define password       "c90222c41416727c4b7569106f8cbfbf6d43b273"


//Logic Preset
#define OFF            0
```

前文中，我们已经对 wifi 的配置信息进行了替换，此处我们需要替换剩下的内容：

- clientIDstr 和 timestamp 若没有改变，不需要修改。
- 三元组内容按照记事本中的内容替换。
- password 使用刚刚生成的哈希值进行替换。

替换完成的效果如下图所示。

```
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BME280.h>
#include <Adafruit_CCS811.h>

//User Modified Part
#define wifi_ssid      "Saxon"
#define wifi_psw       "abcd1234"
#define clientIDstr    "example"
#define timestamp      "999"
#define ProductKey     "algmrulqVfG"
#define DeviceName     "Smart_House"
#define DeviceSecret   "ykPpoww49PxVXDnOimY4tBfWxFIxElZ8"
#define password       "d8304387381ed3acb633a673573951a0f170209f"


//Logic Preset
#define OFF            0
```

单击保存，将修改好的代码烧入开发板。平台的相关配置工作就完成了。

➢ STEP5 查看运行状态

小屋正常运行后，我们可以在物联网平台查看所有的状态信息。

打开物联网平台，在设备管理页面找到新添加的 Smart_House，此时的设备属性应显示为在线。若不显示在线，且小屋上电后发生了响三声的动作，说明烧入了错误的三元组。

单击右侧的查看，进入设备详情页面。

设备信息 Topic列表 运行状态 事件管理 服务调用 设备影子 文件管理 日志服务 在线调试

运行状态 实时刷新 图表 表格

空调状态 查看数据 0 (关) 2019/10/15 10:56:33	蜂鸣器状态 查看数据 0 (关闭) 2019/10/15 10:56:33	窗帘状态 查看数据 1 (开) 2019/10/15 10:56:35	风扇状态 查看数据 0 (关) 2019/10/15 10:56:33
可燃气传感器读数 查看数据 10 2019/10/15 10:56:33	灯光状态 查看数据 0 (关) 2019/10/15 10:56:35	光敏传感器读数 查看数据 174 2019/10/15 10:56:35	水泵状态 查看数据 0 (关) 2019/10/15 10:56:35
室内温度 查看数据 25.53 °C 2019/10/15 10:56:33	土壤湿度传感器读数 查看数据 361 2019/10/15 10:56:35	有机气体浓度 查看数据 8 ppm 2019/10/15 10:56:35	类二氧化碳浓度 查看数据 454 ppm 2019/10/15 10:56:35

页面上方找到运行状态，并勾选实时刷新。此时该页面会实时显示小屋的各种信息。

➢ STEP6 下发静音指令

小屋的蜂鸣器被触发的情况下，可以通过平台下发的静音指令消除报警声音。



在物联网平台左侧的导航栏，监控运维中找到在线调试。



选择设备为 Smart_House，功能为蜂鸣器，将下发消息中的” ”替换为 2。单击发送指令，等待几秒钟即可。

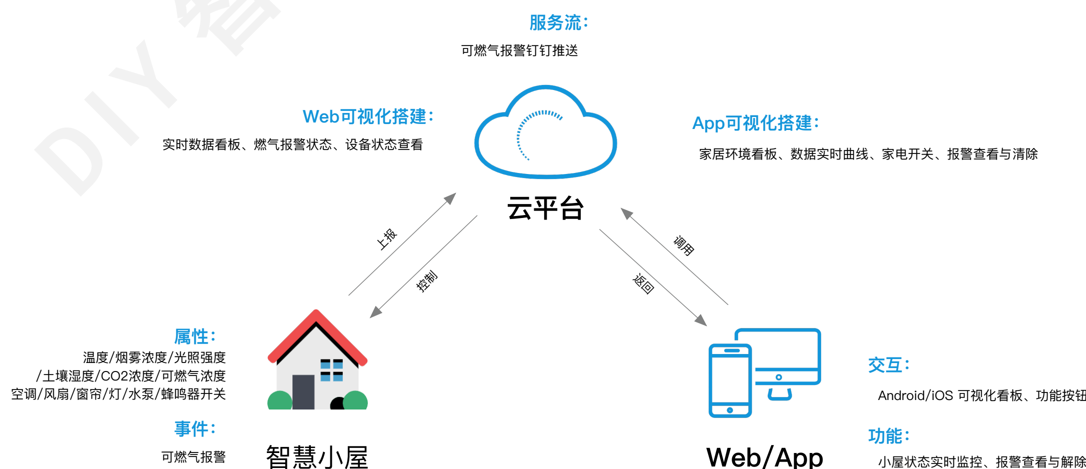
三、平台篇

在前面两个篇章里，相信大家已经初步完成了智慧小屋硬件部分的设计以及嵌入式编码，在本篇章内容将对物联网平台上的操作进行说明，在物联网平台上完成应用层的开发，希望大家在本篇章的学习中掌握物联网平台的基本使用方法，对云端服务、应用前端架构有一定的认识。



图：智慧小屋可视化 Web 页面

在物联网平台的实践部分，设备与云端进行双向的连接，主要通过在云端调用物联网平台提供的设备操作能力或者通过可视化编辑创建服务流完成设备在云端完成对设备的交互逻辑。我们本次实践的设备与阿里云物联网平台简单的交互逻辑如下图所示：

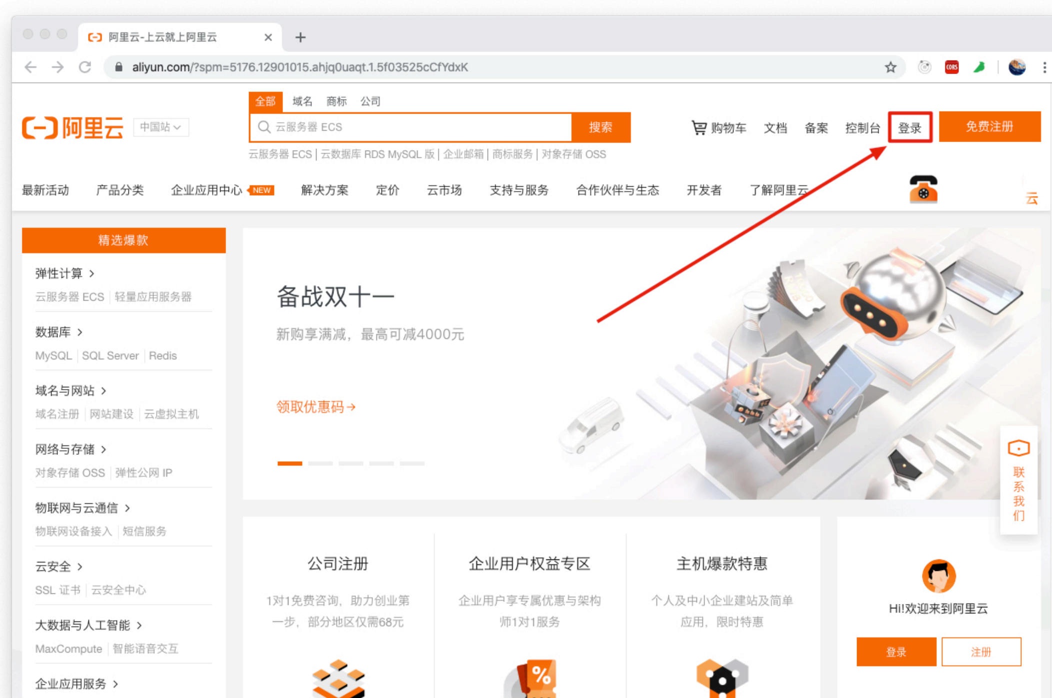


图：智慧小屋端到端交互功能图

如上图所示，智慧小屋设备在运行嵌入式 Arduino 平台上运行的逻辑外，Arduino 还将设备采集到的环境信息、设备的开关状态以及设备检测到可燃气泄露信息进行上报，云平台运行的服务流将会对报警信息进行钉钉推送，并提供 Web 和 App 端的访问能力。

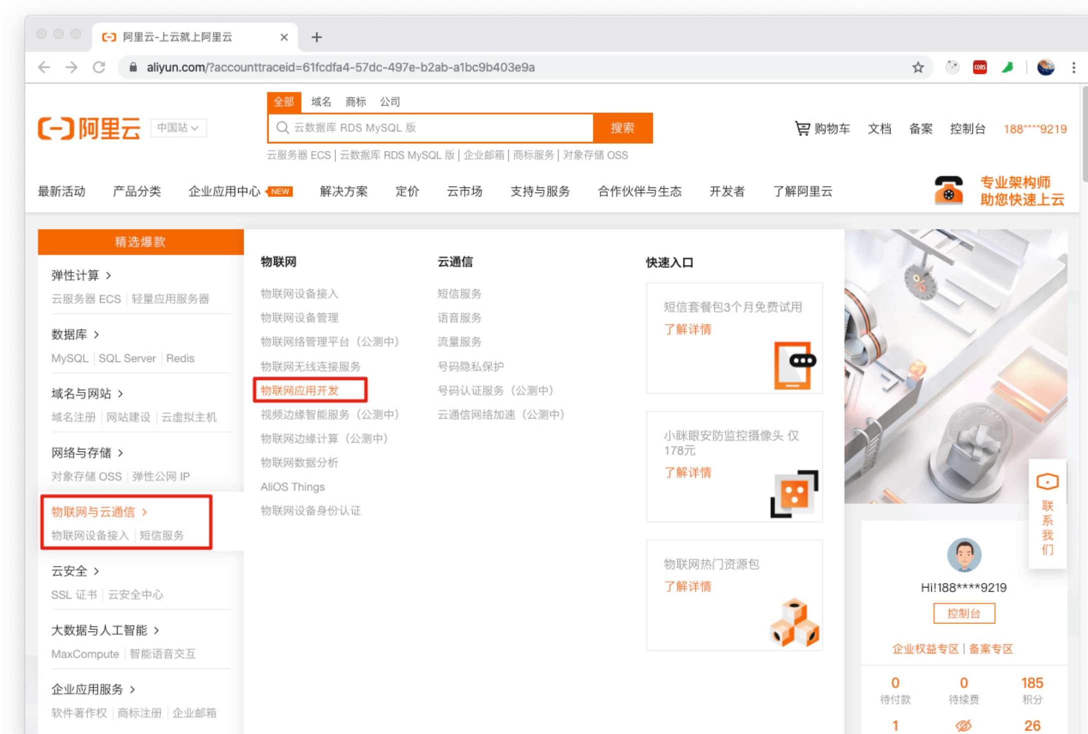
3.1 平台注册与控制台操作

首次使用阿里云物联网平台需要注册阿里云账号，也可以使用支付宝扫码直接登陆，进入阿里云平台，点击右上角登陆，可采用注册阿里云或扫描支付宝二维码完成登录。



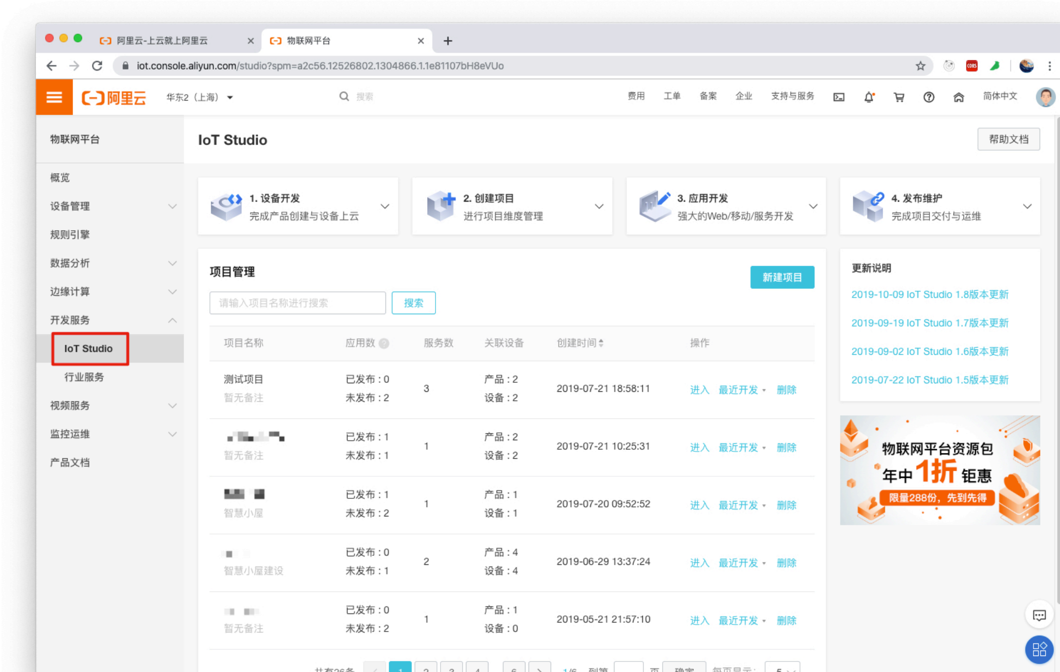
图：登录阿里云平台

完成登录后，依次选择“物联网与云通信”->“物联网应用开发”->“立即使用”进行 IoT Studio 服务的开通。



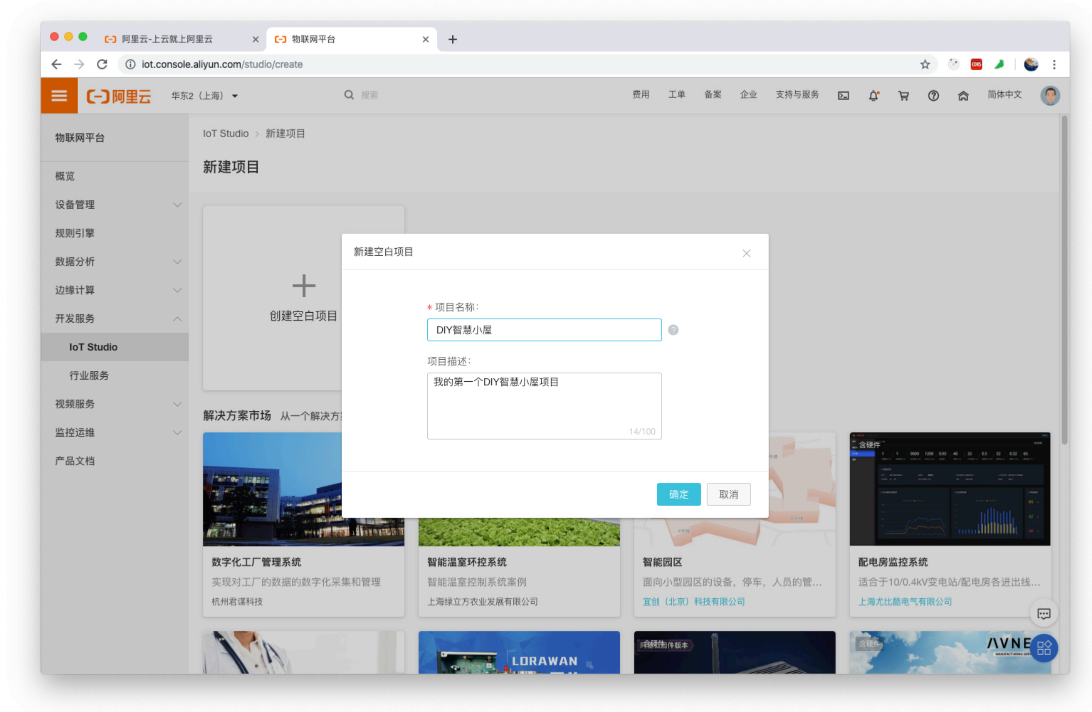
图：在阿里云平台进入物联网应用开发

开通完服务后，进入 IoT Studio 的操作控制台面板，进入后如下图所示，在控制台上显示有 IoT Studio 应用开发流程，即「设备开发->创建项目->应用开发->发布维护」，我们在前面篇章已完成设备开发，直接开始创建项目。



图：物联网应用开发 IoT Studio 控制面板

点击项目管理栏右侧蓝色“新建项目”按钮，进入新建项目流程，我们可以看到 IoT Studio 为开发者提供的模板化项目，可以基于各类模块项目快速初始化项目。在此我们点击创建空白项目（也可基于项目模版创建，学习平台提供标准的项目资源），输入项目名称及备注，平台即会完成项目的创建。



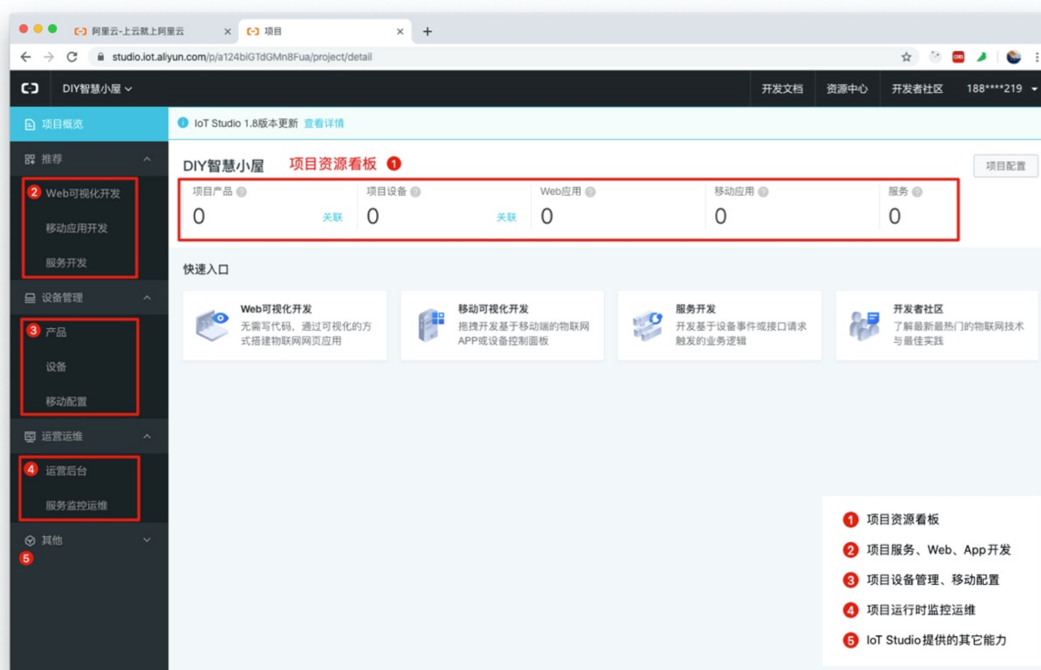
图：创建空白项目

在 IoT Studio 中，所有的资源均以项目制进行管理，项目即可以理解为一个场景的解决方案的集合，各项目之间资源均进行了隔离，项目资源中包括各类产品，基于产品模型创建的设备，编排的云端服务，搭建的 Web 应用以及 App 应用等。

IoT Studio 项目控制台如下图所示，在控制台主要分为 4 个部分，包括项目资源看板、Web/App/服务开发、设备管理、运营中心等，各个部分主要功能：

- 项目资源看看板：项目下资源数据统计，包括项目内产品数量、设备数量、Web 应用数量、移动应用数量、服务数量。
- Web/App/服务开发：提供了 IoT Studio 核心开发能力，即设备上云后服务编排，Web 搭建、App 搭建功能。

- 设备管理/移动配置：提供 IoT Studio 内的产品管理、设备管理以及 IoT Studio 到物联网平台的设备关联导入，提供移动 App 配网引导配置以及面板配置。
- 运营运维：对项目管理各类角色、管理员进行管理，以及服务的权限配置与管理能力。服务监控运维中可以对已发布的服务进行监控，如调用次数、错误次数、运行时长统计。



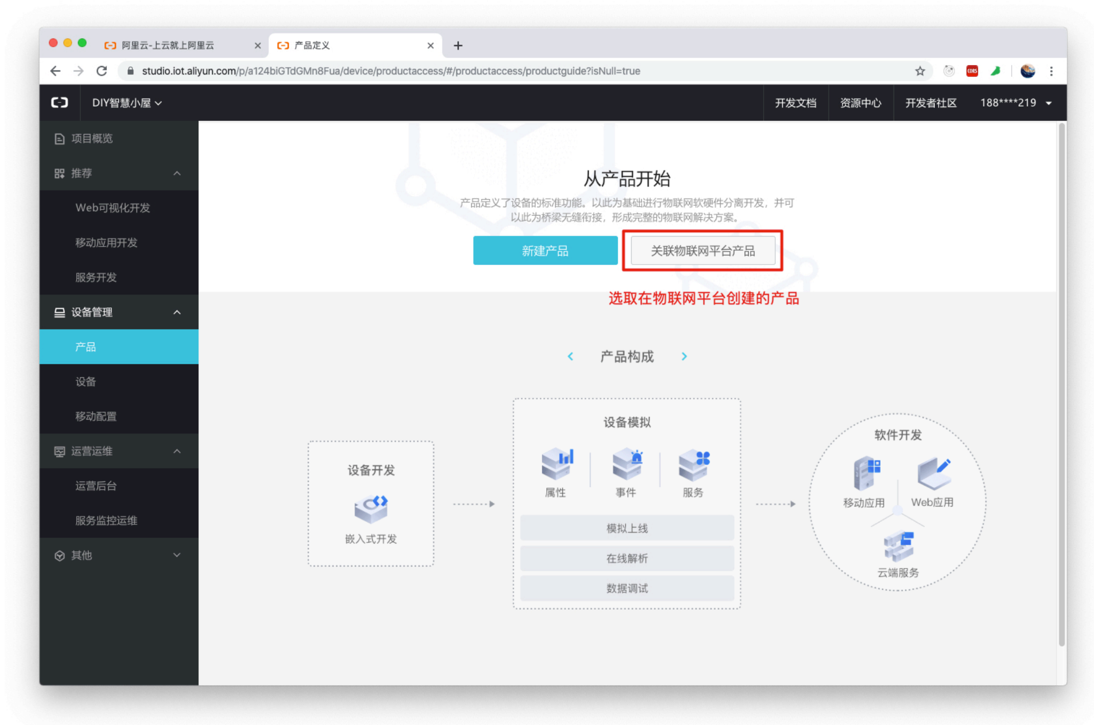
图：IoT Studio 项目控制台面板

在本小节，已经完成平台注册登陆，并且在 IoT Studio 中创建了第一个 DIY 智慧小屋项目，完善项目时，我们需要完成设备绑定与调试、服务流编排、Web/App 开发这几个阶段，并在各个阶段，我们可以依赖 IoT Studio 提供的丰富调试能力对各个阶段完成调试。

3.2 设备绑定与调试

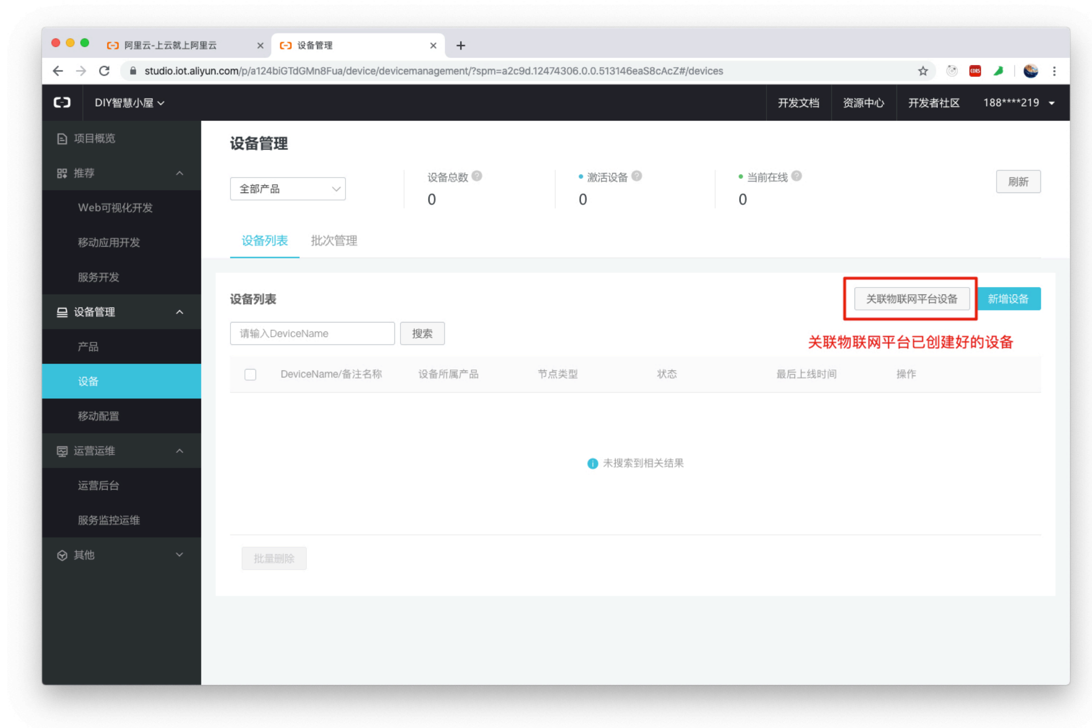
新创建的项目中是不包含产品和设备，可以通过在 IoT Studio 项目中直接创建、直接从物联网平台导入两种方法，在嵌入式开发篇章，我们已经在物联网平台完成了产品的创建和设备调试，在此我们直接从物联网平台导入产品和设备即可。

在项目控制台，选择左侧设备管理->产品，选择关联物联网平台产品，在列表中选择“关联物联网平台产品”即可，如下图所示：



图：IoT Studio 中关联物联网平台产品

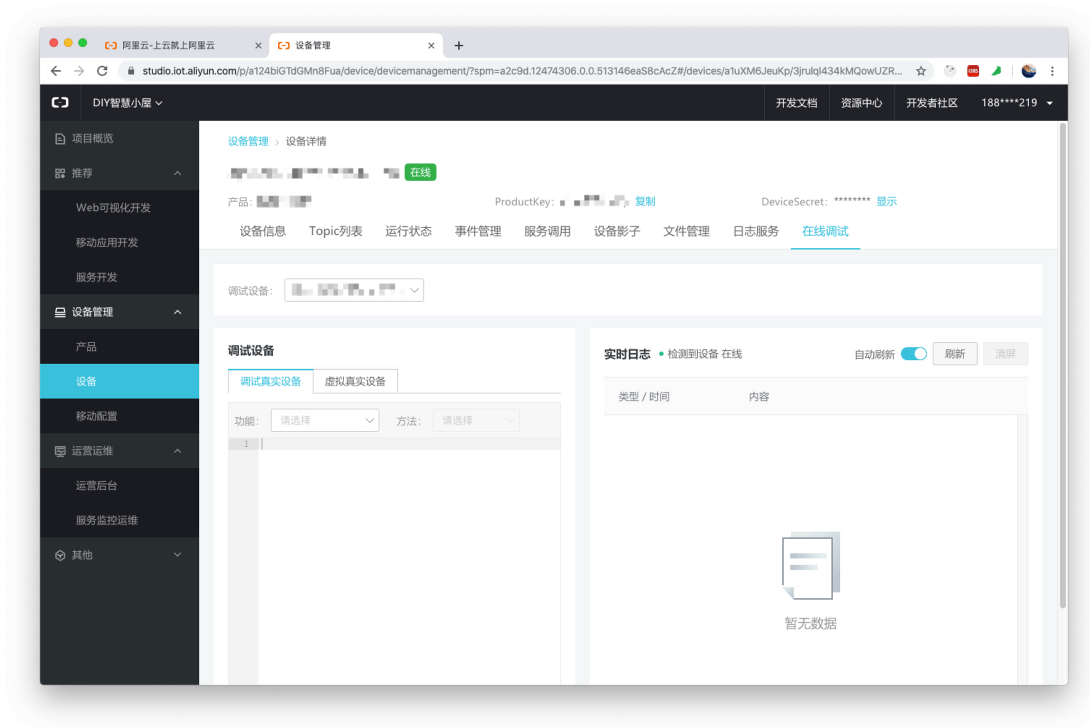
关联产品后，有两种方法创建设备，可通过 IoT Studio 中的设备管理直接基于导入的智慧小屋直接新建智慧小屋设备，或者在设备管理直接导入物联网平台的设备，这里我们可以直接导入嵌入式开发时调试好的智慧小屋设备，少去重新烧写设备三元组固件的流程。



图：关联物联网平台设备

如上图所示，在设备管理页面点击“关联物联网平台设备”，在弹出的列表中勾选智慧小屋产品下已调试完成的设备，完成设备导入。

至此，我们已经完成产品和设备的导入，IoT Studio 中也提供了对设备进行调试的能力，可以观察设备的上行数据，以及对设备进行下行命令控制。IoT Studio 同时提供虚拟设备，如果开发者手中没有真实硬件设备，也可以在虚拟设备中模拟设备数据上报，解耦硬件开发和后续服务、Web、App 搭建。

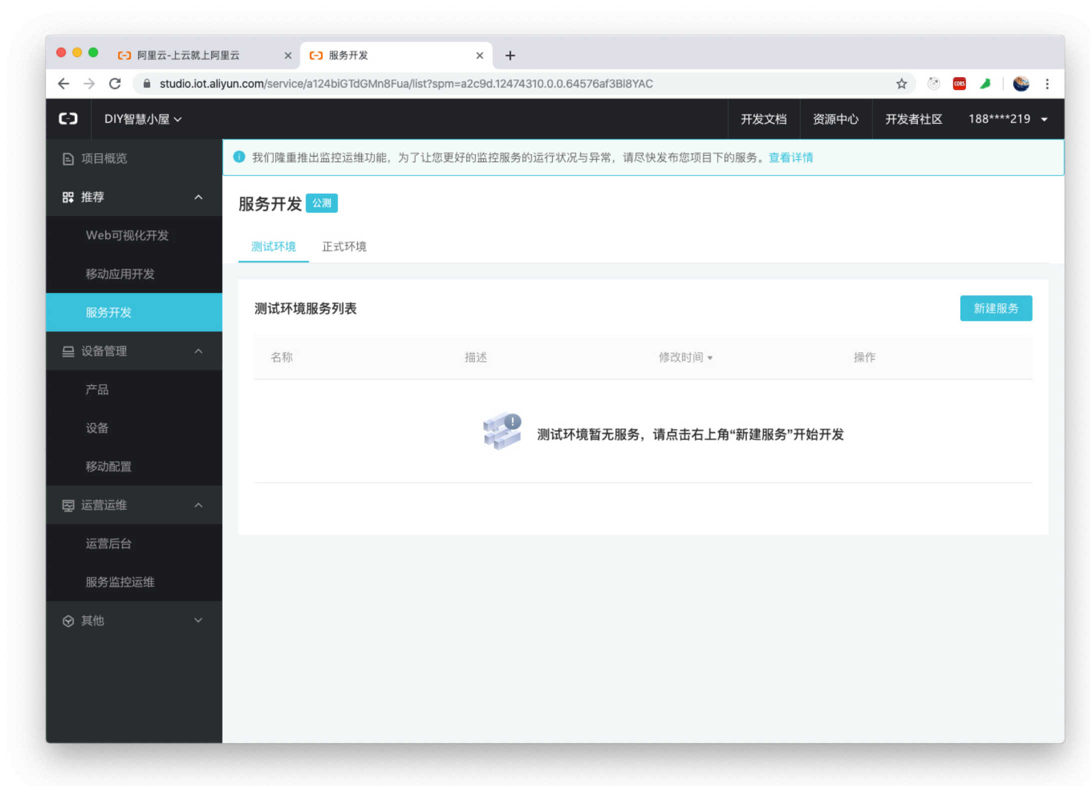


图：IoT Studio 中设备调试窗口

3.3 服务编排

在上一小节，我们在 IoT Studio 平台上完成了设备的导入以及调试，设备在数据上云后，需要依靠服务完成数据的各类处理，例如数据存储、数据加工、消息推送等。

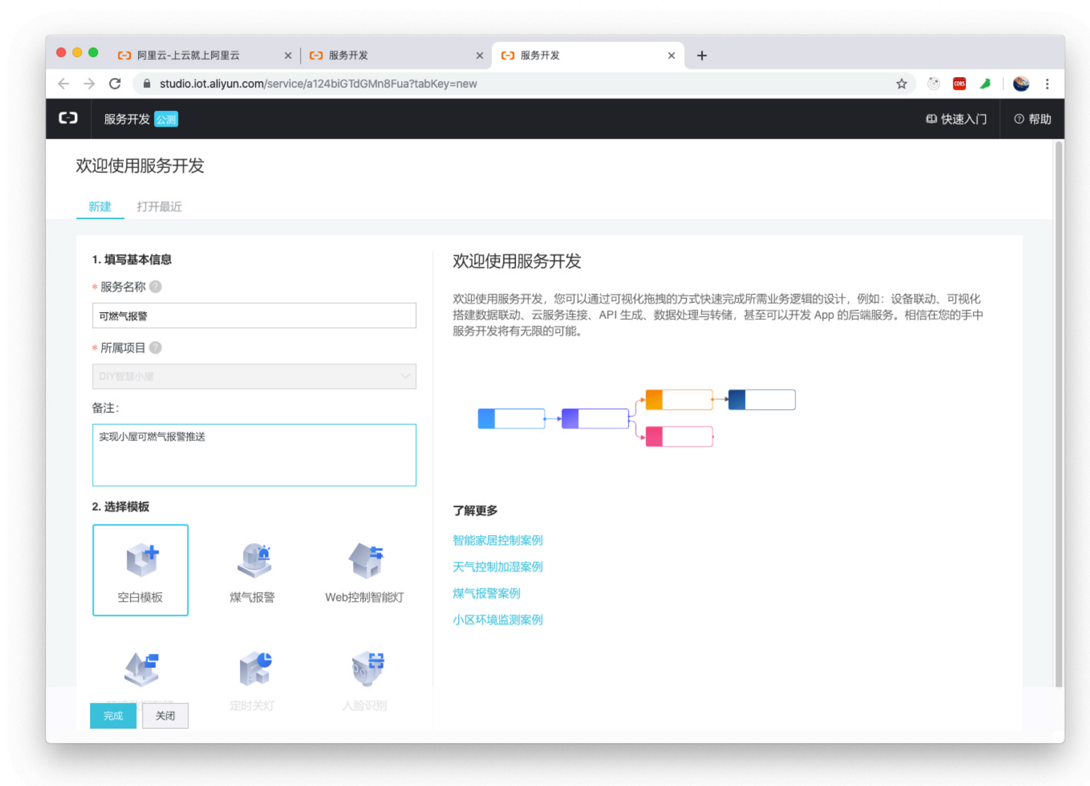
在 IoT Studio 项目控制台，点击左侧菜单“服务开发”进入服务开发控制台，如下图所示：



图：服务开发控制台

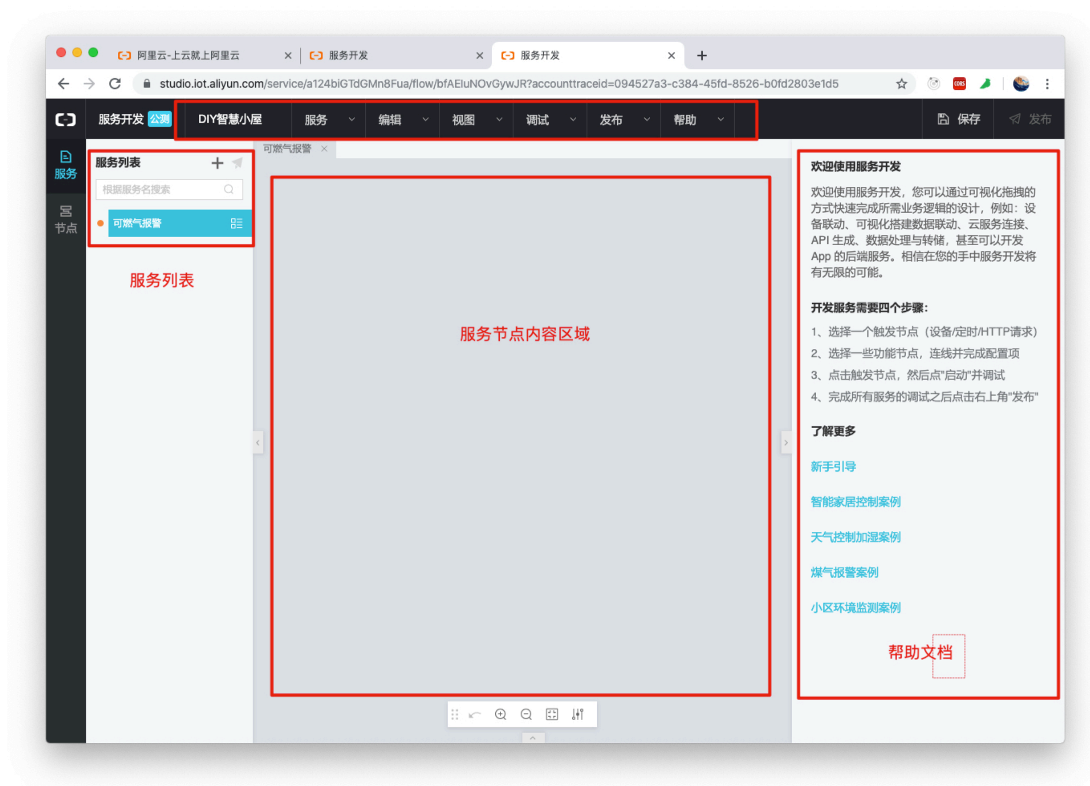
服务开发具有两套开发环境，如上图所示，分为“测试环境”和“正式环境”，一般来说，服务开发具有调试阶段（开发时）和运行阶段（运行时），对于已发布的服务将稳定运行在运行时的环境，服务开发的测试环境和正式环境有利于将开发时和运行时环境解耦开，尽量减少对已上线稳定运行的服务的干扰。

点击“新建服务”创建 DIY 智慧小屋第一条服务流。与项目模版类似，在服务开发阶段，IoT Studio 也提供了服务流模版，包含报警、设备控制、数据存储等，可以帮助开发者快速基于模版进行构建。本次基于空白模版进行创建，输入服务名称以及备注，如下图所示：



图：创建服务

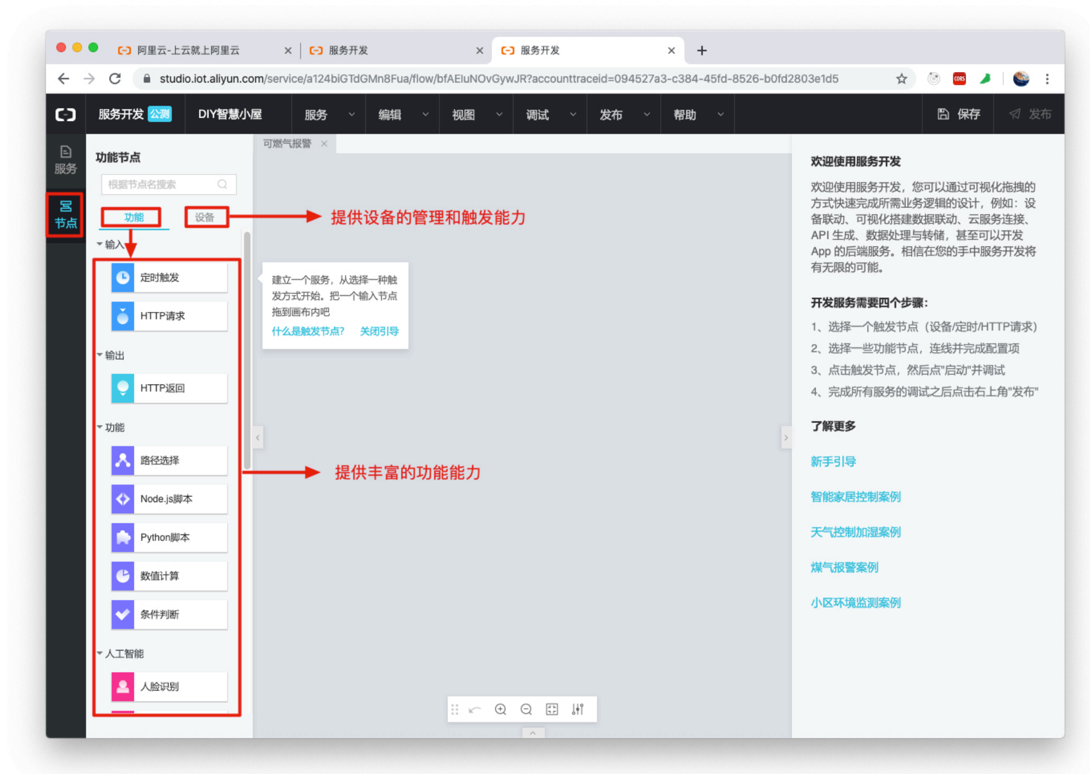
创建好服务后，即会进入此项目下的服务列表，我们可以点击加号继续创建服务，也可以单击服务右侧图标对服务进行编辑和删除操作，服务列表可以在多个服务间快速跳转，进行编辑。在服务列表右侧为帮助栏，提供当前服务工作台的一些指导，建议阅读。在服务中间的区域为服务节点内容的编辑区域（由于未对服务编辑下图编辑区域为空），在上方顶栏提供服务的编辑功能，如保存、新建、调试、发布等等。



图：IoT Studio 服务工作台功能区划分

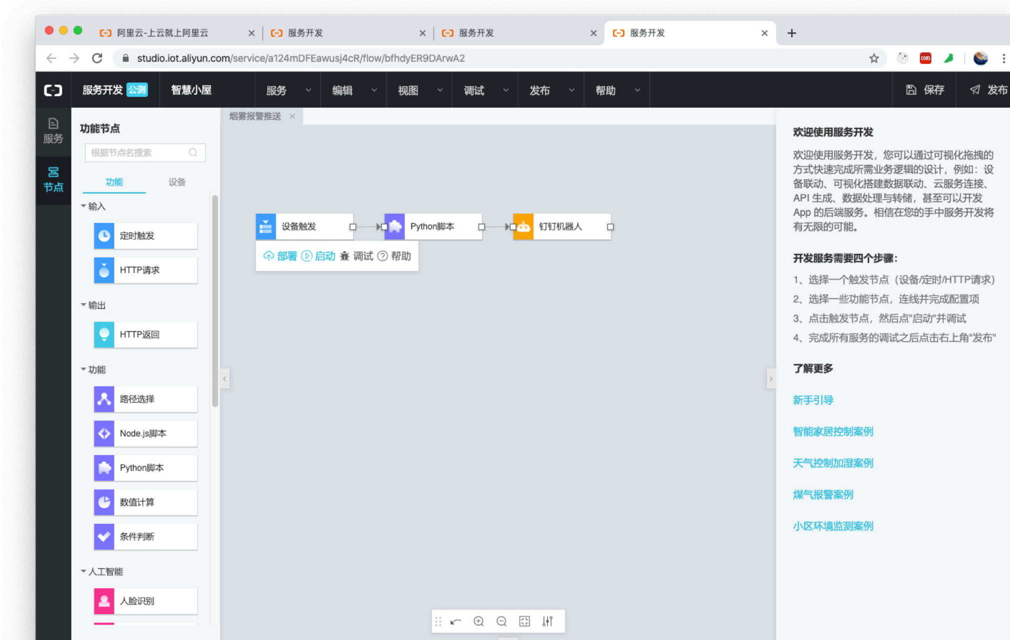
从左侧“服务”切换到“节点”，我们即可使用 IoT Studio 提供的丰富节点对服务进行编辑与完善，节点共分为输入、输出、功能、AI、消息、API、存储几类，在每个分类中均有具体的功能节点提供相应的能力。

在设备区域，将会展示导入到此项目的产品和设备信息，提供设备触发和对设备的控制能力。



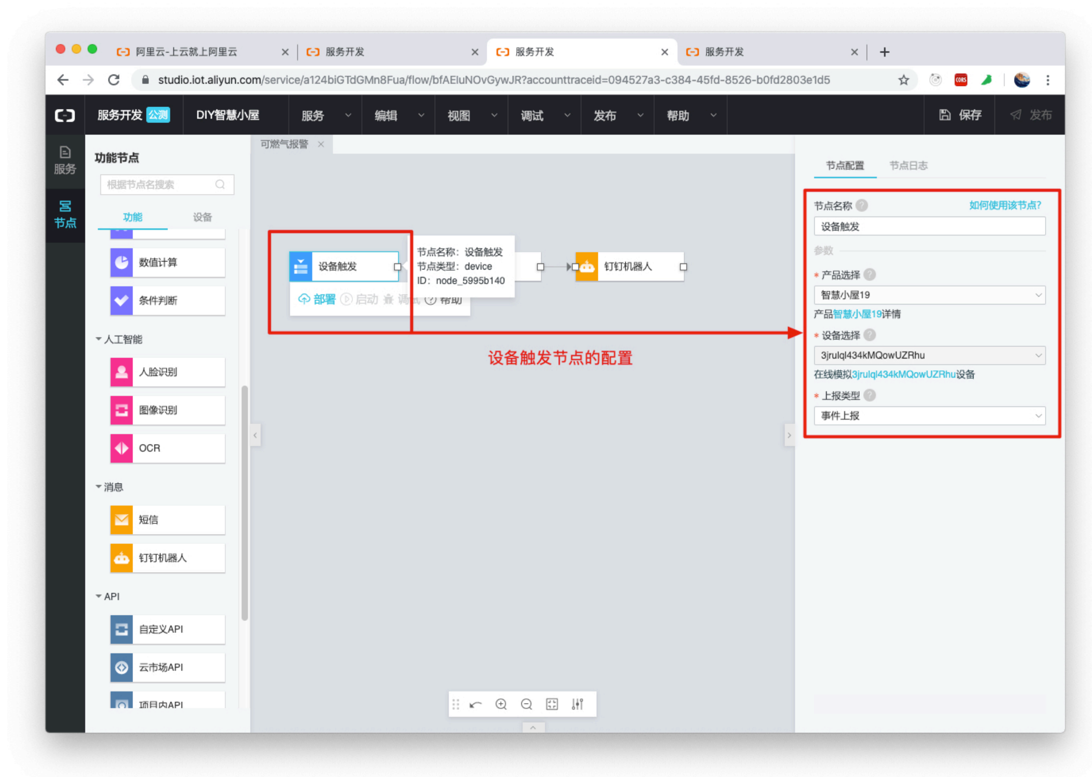
图：IoT Studio 提供的节点能力

我们通过节点组合，创建“设备触发->Python 脚本->钉钉机器人”的一条服务流，实现设备上报可燃气报警事件的消息钉钉推送，如下图所示：

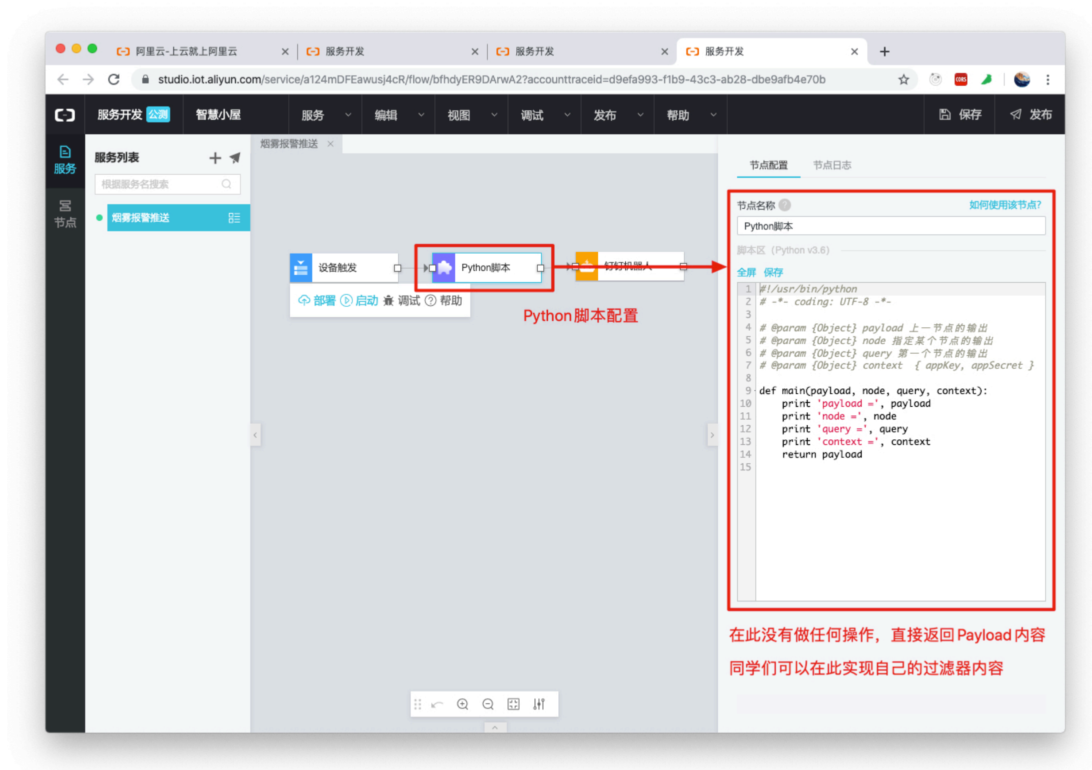


图：创建设备触发服务流

这条服务流的运行过程如下：在以上三个功能节点中，设备触发节点可配置为智慧小屋设备，触发类型为事件触发，即当小屋设备开始上报事件时，设备触发节点会激活此服务流，将设备上报的事件数据传递到 Python 脚本节点进行处理，Python 节点处理后的数据内容再发送给钉钉机器人节点进行消息的推送。

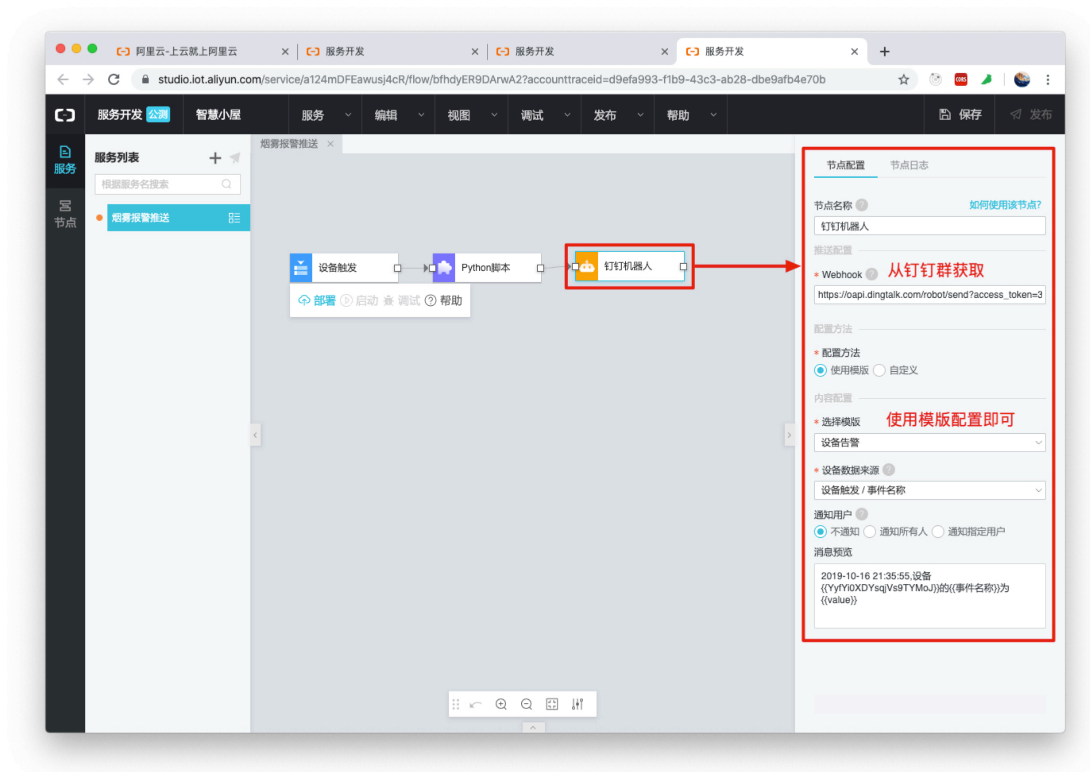


图：设备触发节点的配置



图：Python 脚本配置

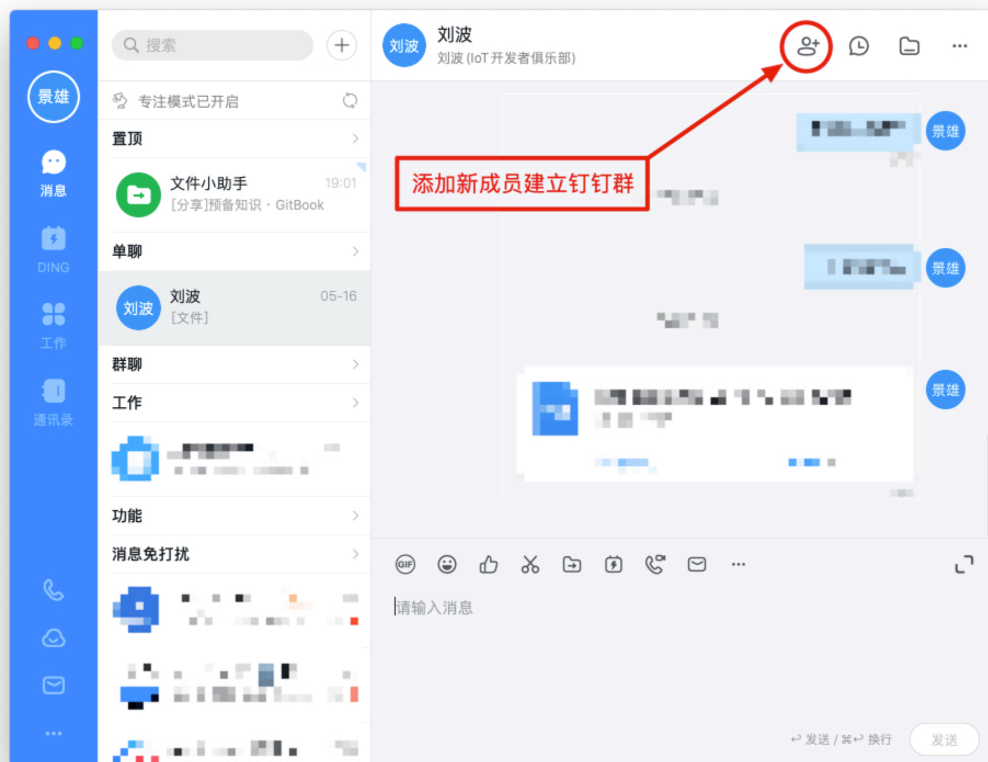
Python 脚本配置如上图所示，Python 脚本在两个节点之间实际上起“过滤器”的效果，可以对上节点传入的 Payload 内容做解析和处理，同时将需要传递给下游节点的消息通过 return 进行传递，在智慧小屋可燃气报警的场景中，由于我们已经在设备触发节点配置触发类型为事件触发，所以 Python 节点没有对数据进行任何处理即返回（在此实际 Python 节点去除服务也可正常运行），Python 节点在此可以作为一个过滤器来使用，通过解析 Payload 内容，对数据进行加工处理再传递到下游节点。感兴趣的同学如果对 Python 比较熟悉，可以尝试添加自己的处理逻辑上去。



图：钉钉机器人配置

钉钉机器人需要配置 Webhook 以及配置消息内容。Webhook 是钉钉开放平台进行钉钉消息推送所需要的唯一 ID 地址，需要在钉钉群中进行开通钉钉机器人即可获得，具体操作过程如下：

1. **创建钉钉群：**开发者需要注册钉钉账号，可以下载手机版钉钉进行注册，并加取至少两位钉钉好友创建钉钉群，创建钉钉群只能通过电脑版钉钉进行添加，还需下载钉钉电脑版进行登录，选择一位钉钉好友进入聊天窗口，点击右上角添加新的聊天成员创建钉钉群。



刘波
刘波 (IoT开发者俱乐部)
添加新成员建立钉钉群
搜索
专注模式已开启
置顶
文件小助手
[分享]预备知识 · GitBook
单聊
刘波
[文件]
群聊
工作
功能
消息免打扰
请输入消息
发送



钉钉消息通知群
测试
点击添加钉钉机器人
新建钉钉群成功
搜索
专注模式已开启
置顶
文件小助手
[分享]预备知识 · GitBook
单聊
刘波
[已读] [文件]
群聊
钉钉消息通知群
[任务]2019-07-21 19:5...
工作
功能
消息免打扰
请输入消息
发送

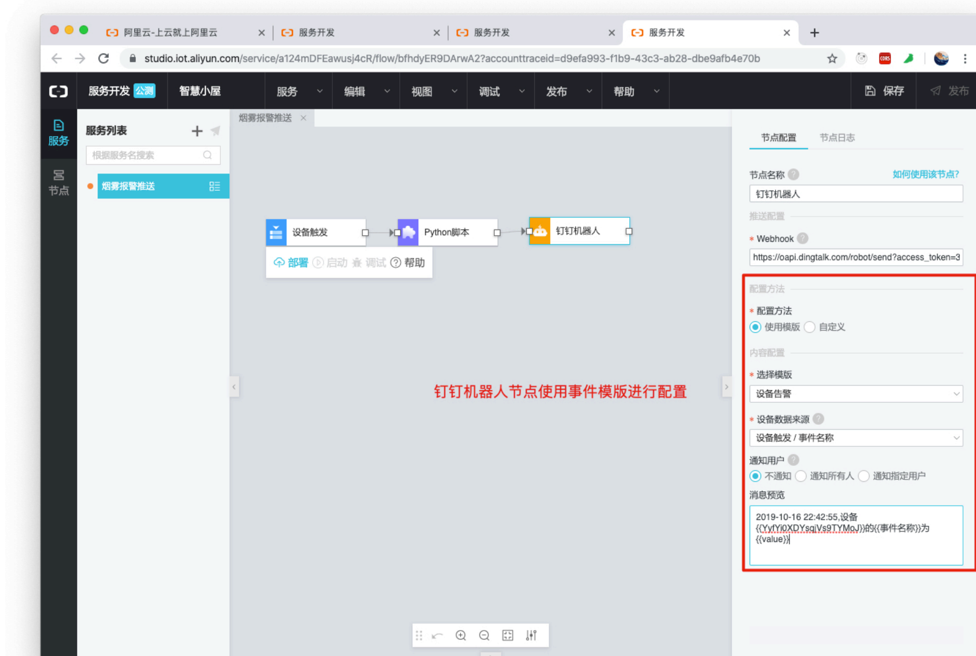
图：通过添加成员建立钉钉群

2. **添加群机器人：**在钉钉群右侧点击更多设置->智能群助手->添加更多->添加自定义机器人，复制生成的 Webhook 地址信息，并在 IoT Studio 钉钉机器人节点粘贴。



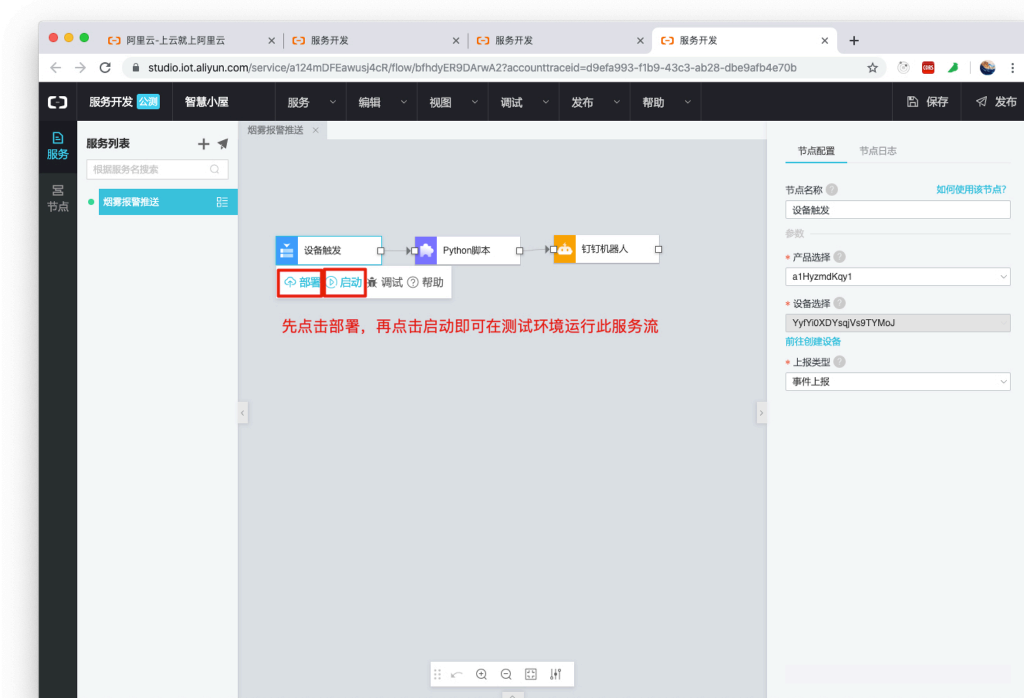
图：添加自定义机器人

完成钉钉机器人 Webhook 添加后，在钉钉机器人节点配置方法选择配置模版，并使用平台提供的设备告警模版->设备触发/事件名称，在下方可以预览推送的消息内容。



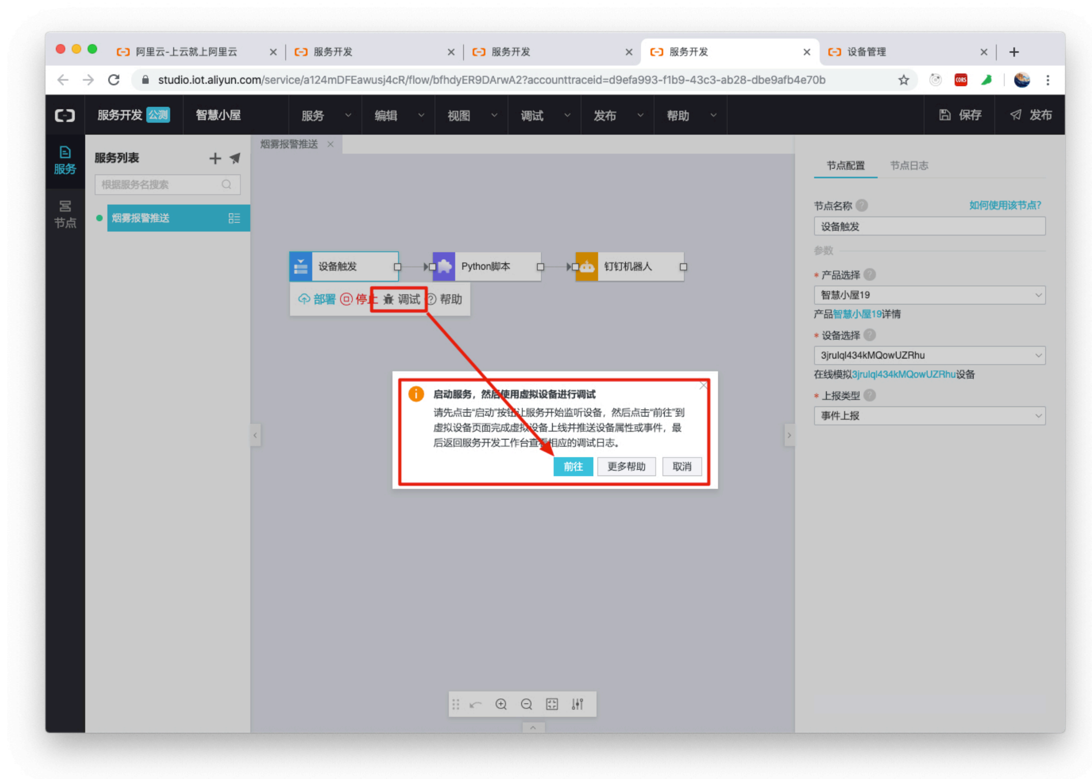
图：配置钉钉机器人节点文案内容

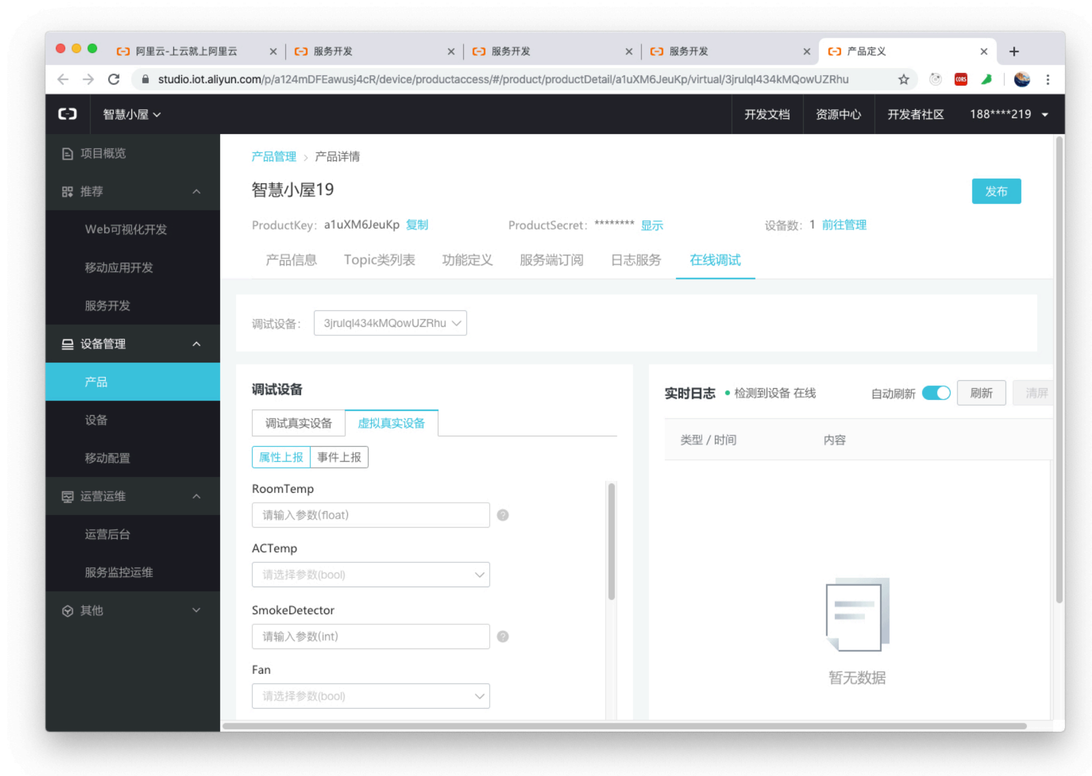
完成文案配置后，即可对整条服务流进行部署与调试，服务流的部署和调试均在测试环境中运行，不会影响到已发布的服务。



图：部署和启动服务流

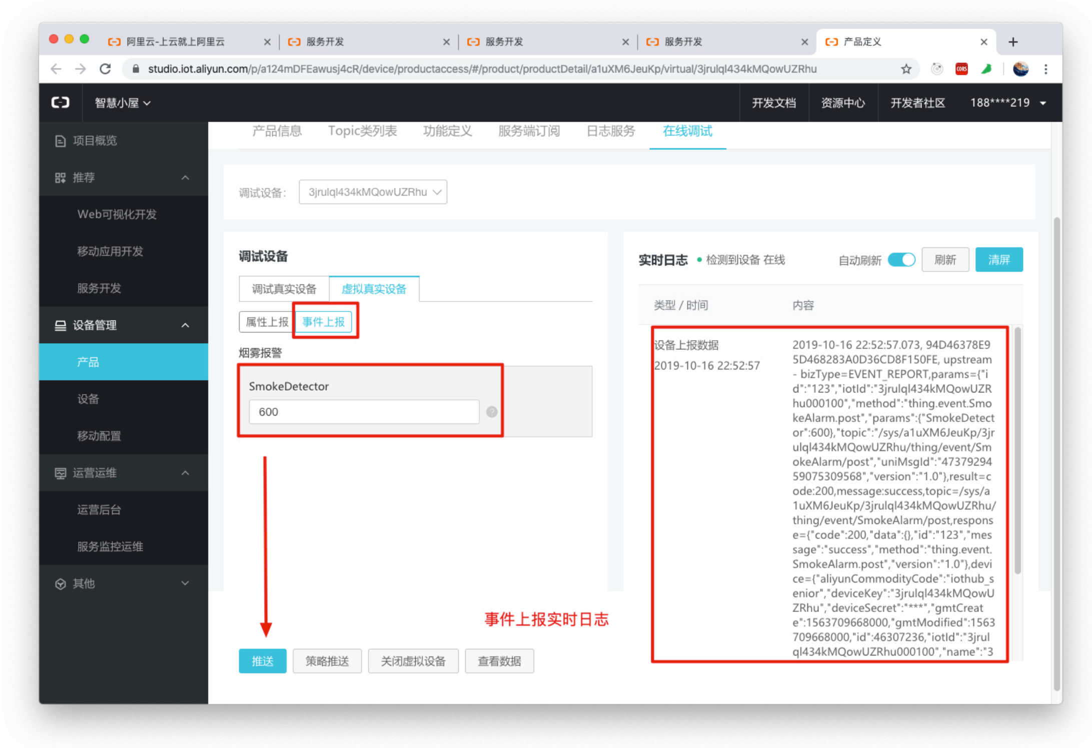
在测试时，既可以使用真实的智慧小屋嵌入式设备进行调试，也可以使用虚拟设备进行调试，在此演示使用虚拟设备调试的过程。在部署启动服务后，点击设备触发节点的调试按钮，即会跳出弹窗提醒开启虚拟设备调试，点击前往即会进入到 IoT Studio 的设备调试中。



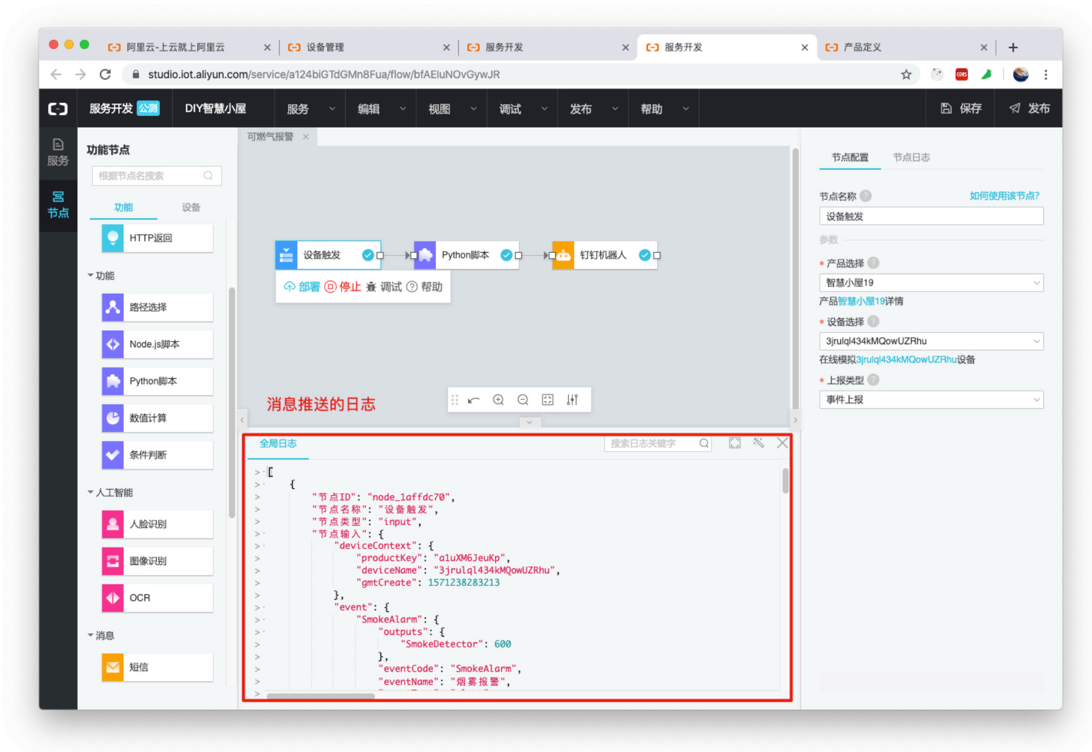


图：进入虚拟设备在线调试

在调试时，我们模拟设备上报可燃气报警事件，观察服务流的运行情况，在虚拟设备窗口选择“事件上报”，填写事件上报的浓度参数，点击推送，即会在实时日志中查看到设备上报的消息日志。



图：推送虚拟设备数据



图：在服务流编辑区域查看到节点日志



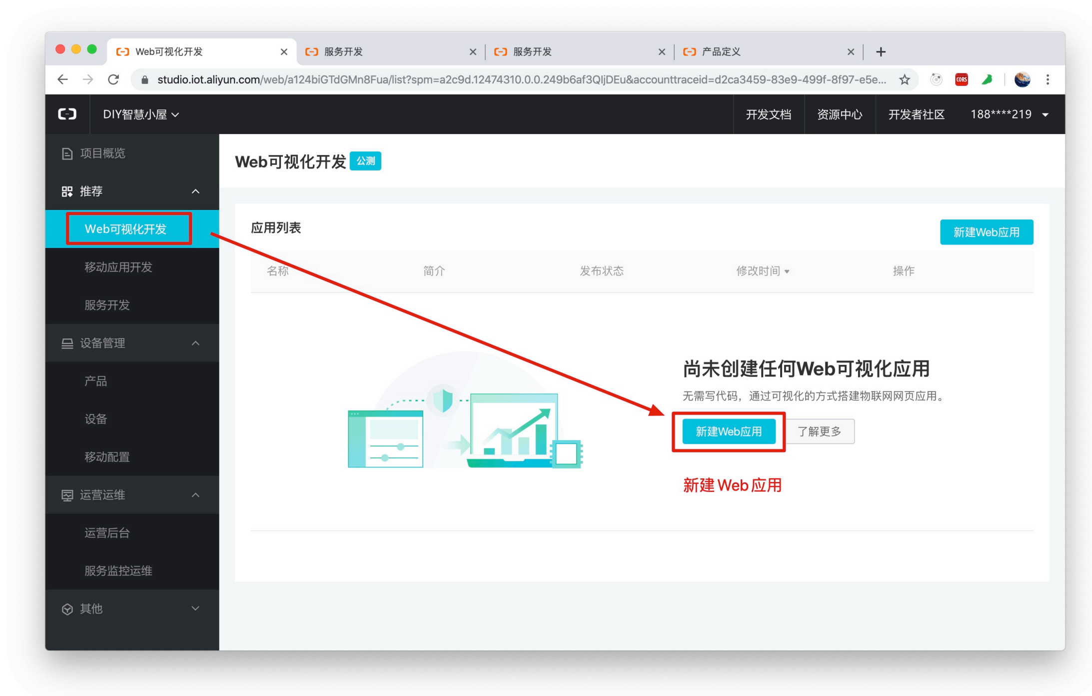
图：接收到钉钉机器人的推送

至此，已经完成了钉钉机器人推送可燃气报警事件的服务流，通过快速拖拽配置即可上线，同学们还可以尝试使用其它功能节点组合成各类丰富的场景功能，完善小屋的场景体验。

3.4 Web 可视化搭建

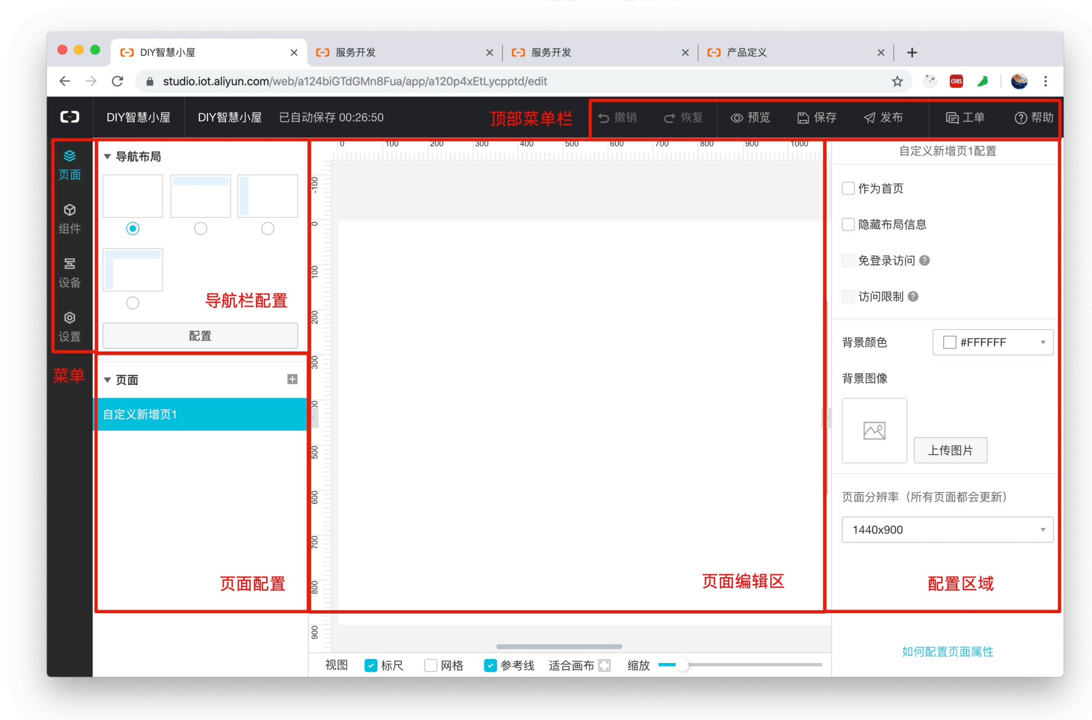
智慧小屋这类家居场景中往往会有一块智能家居看板，用户可以通过看板了解到家居的环境状况和设备的运行情况，在看板上对设备直接进行控制与操作。作为直接与用户交互的大屏，可以通过多种开发途径实现，在本小节将介绍 IoT Studio 提供的 Web 可视化搭建能力，开发者只需要对各类组件进行拖拽和配置即可完成 Web 应用的搭建，使用起来非常方便。

在项目控制台的“推荐”菜单栏中选择“Web 可视化开发”，新建 Web 可视化应用，进入 Web 可视化开发工作台。



图：新建 Web 应用

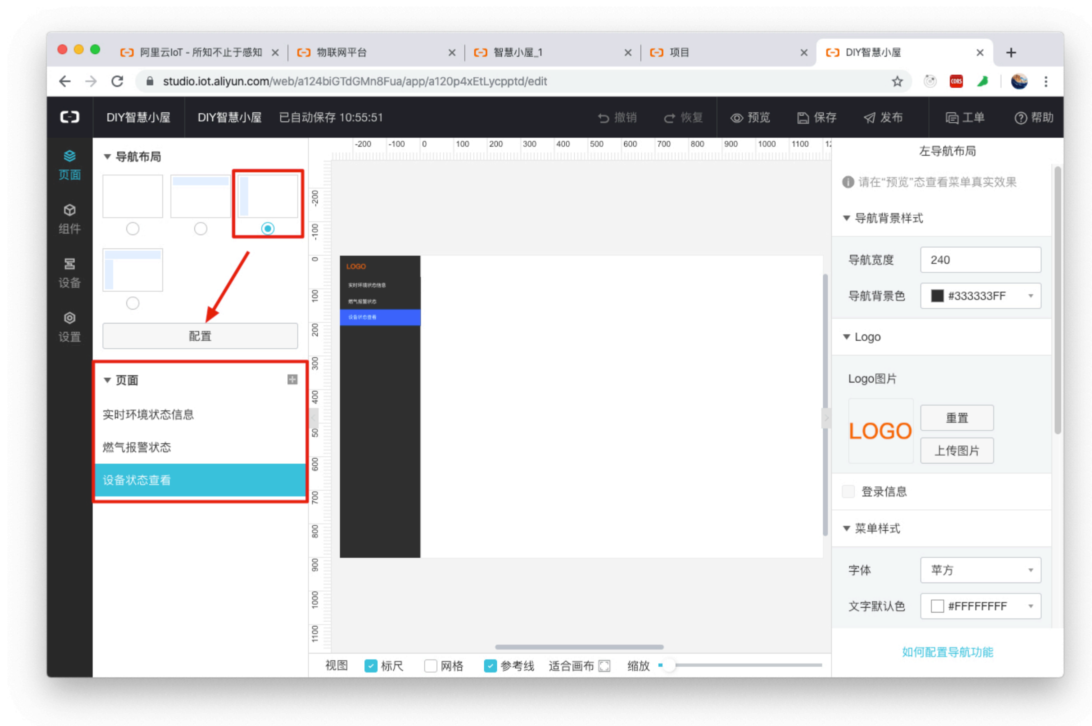
Web 可视化开发工作台页面功能区划分如下图所示：



图：Web 可视化开发工作台功能区划分

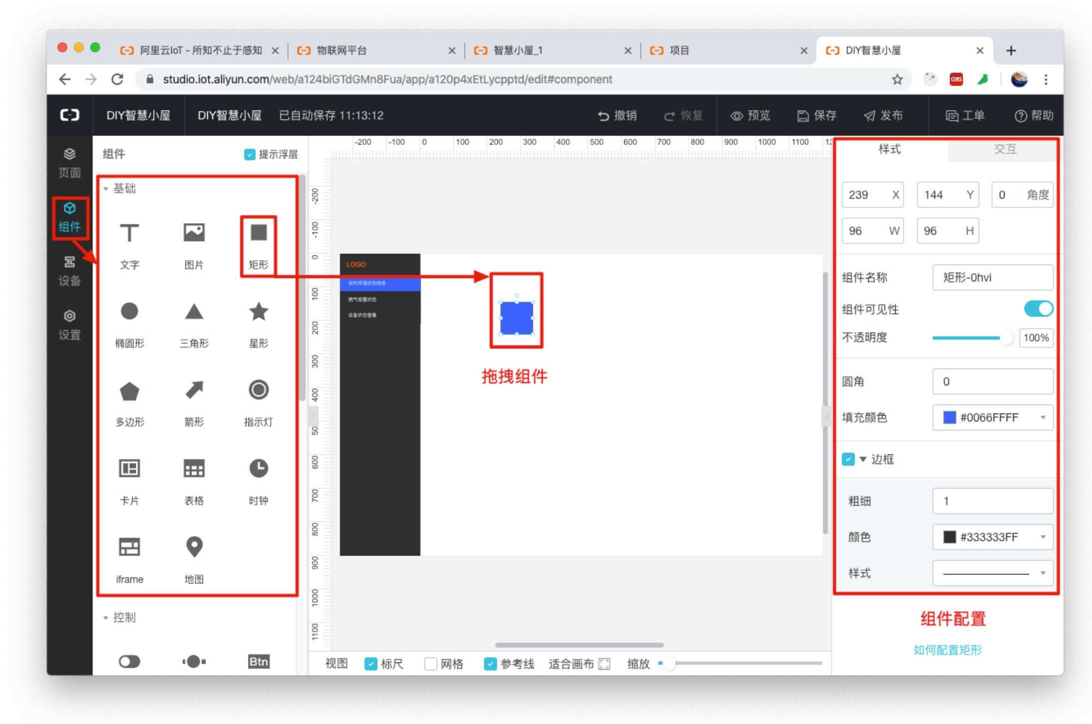
新建三个页面，分别是“实时环境状态信息”页面，用于展示智慧小屋的环境情况；“燃气报警状态”页面，实时监测燃气报警状态，并可下发指令关

闭燃气报警蜂鸣器；“设备状态查看”页面，可查看与管理智慧小屋产品下的设备。新增页面完成后，在页面上方选择导航布局，点击下方配置即可自动生成页面菜单。



图：新建页面生成导航菜单

选中具体页面，在左侧栏选中组件，即可拖拽左侧组件到右侧编辑区域进行配置。

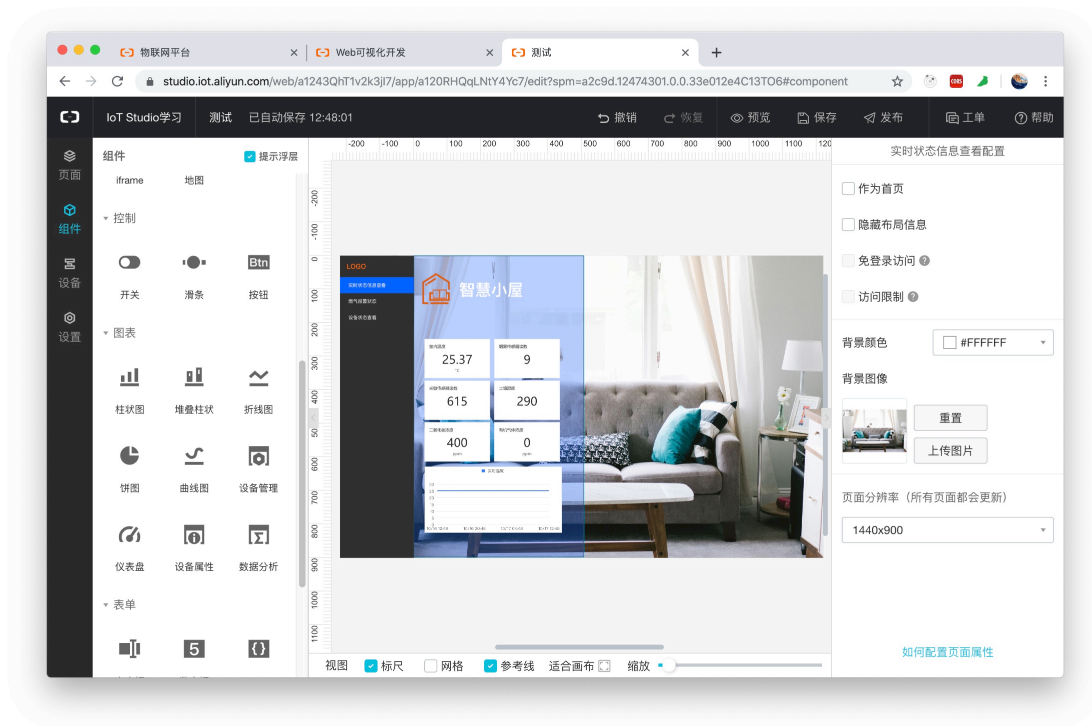


图：组件拖拽配置

Web 可视化开发工作台提供的组件有：

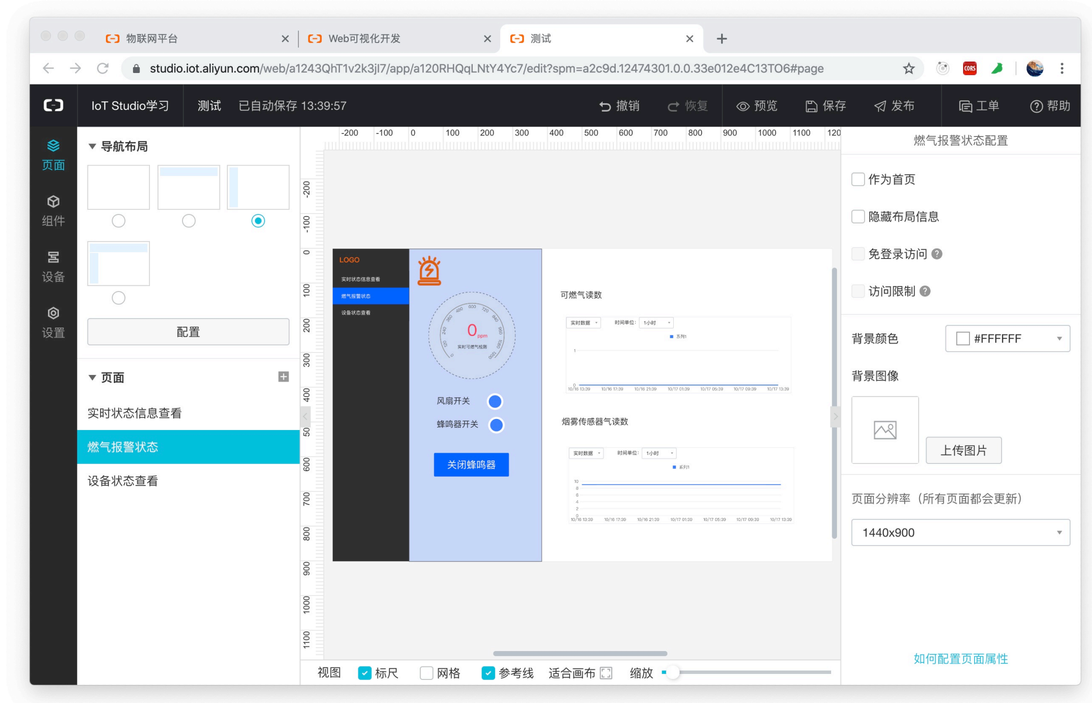
- 基础组件：包含文字、图片、各类图形、卡片、指示灯、表格、框架 iframe、地图组件等。基础组件是在 Web 可视化搭建过程中使用最多的组件种类。
- 控制组件：包含开关、横条、按钮组件，可以定义丰富的交互功能。（基础组件中交互事件也具有按钮组件相同的功能）。
- 图表组件：包含各类统计图表类型，并提供了设备管理、设备属性和数据分析的功能块，可以快速集成这些能力。
- 表单组件：表单组件为用用户提供输入和输出的组件类型，包含文本框、数字框、JSON、下拉框、时间选择等。

在了解了各个组件的各个功能后，我们通过拖拽组件开始组合 DIY 智慧小屋的三个页面。小屋实时环境状态信息页面如下所示，其中使用到的组件有图片、卡片、文字、矩形、曲线图，其中 6 个卡片组件分别显示设备上报的室内温度、烟雾传感器读数、光敏传感器读数、土壤湿度、二氧化碳浓度、有机气体浓度，下方实时曲线图表显示实时室内温度的变化曲线。当配置组件的数据关联到设备时，可以在编辑区域实时观察到设备的状态变化。



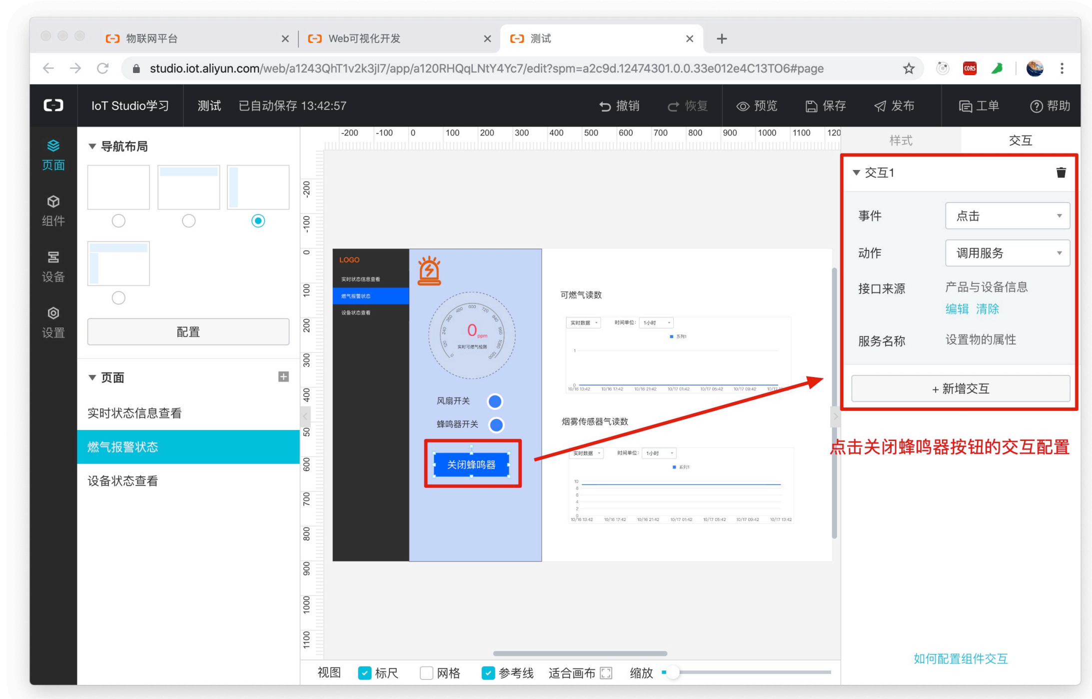
图：智慧小屋实时状态查看页面

接下来我们完善燃气报警状态页面，使用到的组件有矩形、图片、仪表盘、指示灯、文字、按钮、曲线图组件，样式如下图所示：



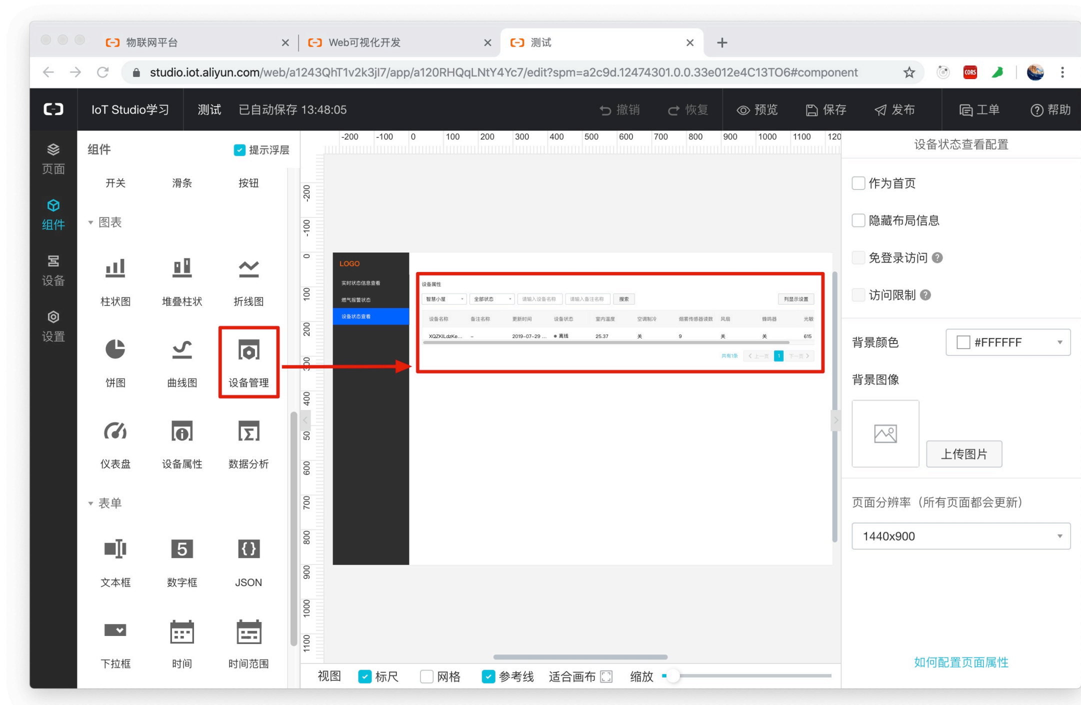
图：燃气报警状态页面

其中按钮组件配置功能为关闭蜂鸣器，通过配置按钮的交互为调用服务，设置物的属性实现，点击关闭蜂鸣器按钮时，将会下发设置蜂鸣器开关为关闭状态的消息给智慧小屋设备。



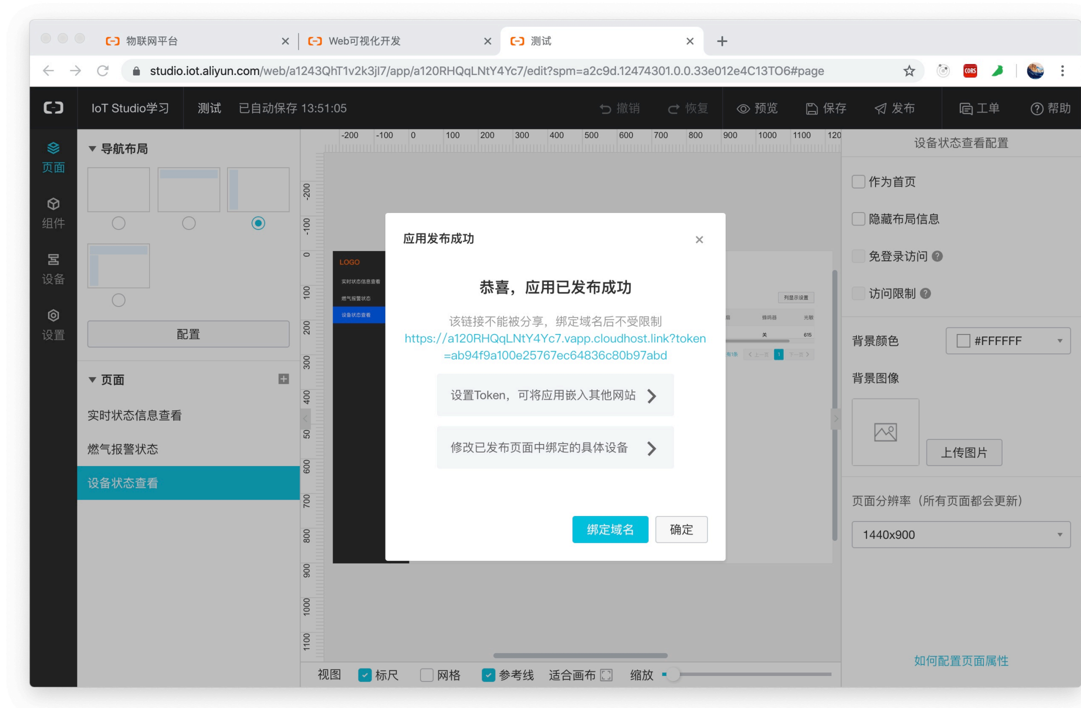
图：关闭蜂鸣器按钮的交互配置

设备状态查看页面，我们来体验 Web 可视化工作台提供的设备管理组件即可，直接将设备管理组件拖拽进页面。设备管理组件直接通过表格展示产品下的设备列表和设备的一些状态，如果在此产品下有多台设备，在设备管理组件中即可查询到多条记录，并可以实时显示设备的在线状态以及设备的属性信息。



图：设备状态查看页面

在三个页面都完成开发后，可以点击上方的预览对页面进行实时的查看与调试，点击发布即可将此页面发布此应用的一个版本，发布后的页面可以绑定域名或添加访问认证，同学们可以查看平台提供的帮助文档。



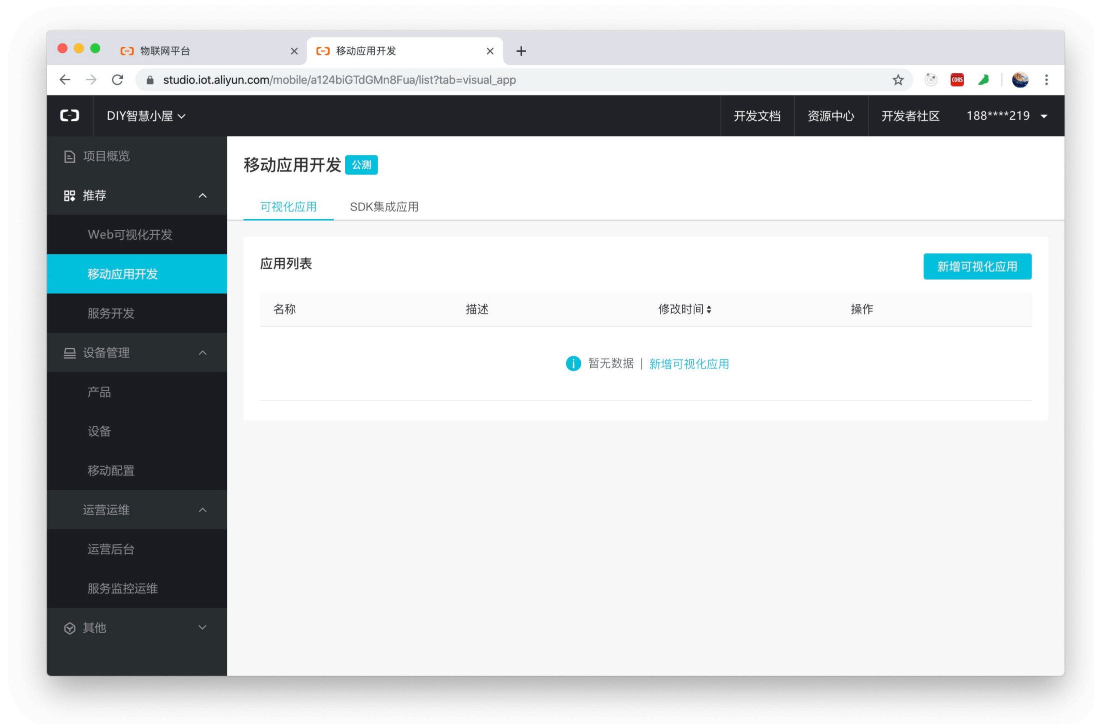
图：发布 Web 可视化应用

至此，我们已经完成 Web 可视化开发的操作，再下一小节 App 可视化搭建也提供了与 Web 类似的可视化搭建能力。

3.5 App 可视化搭建

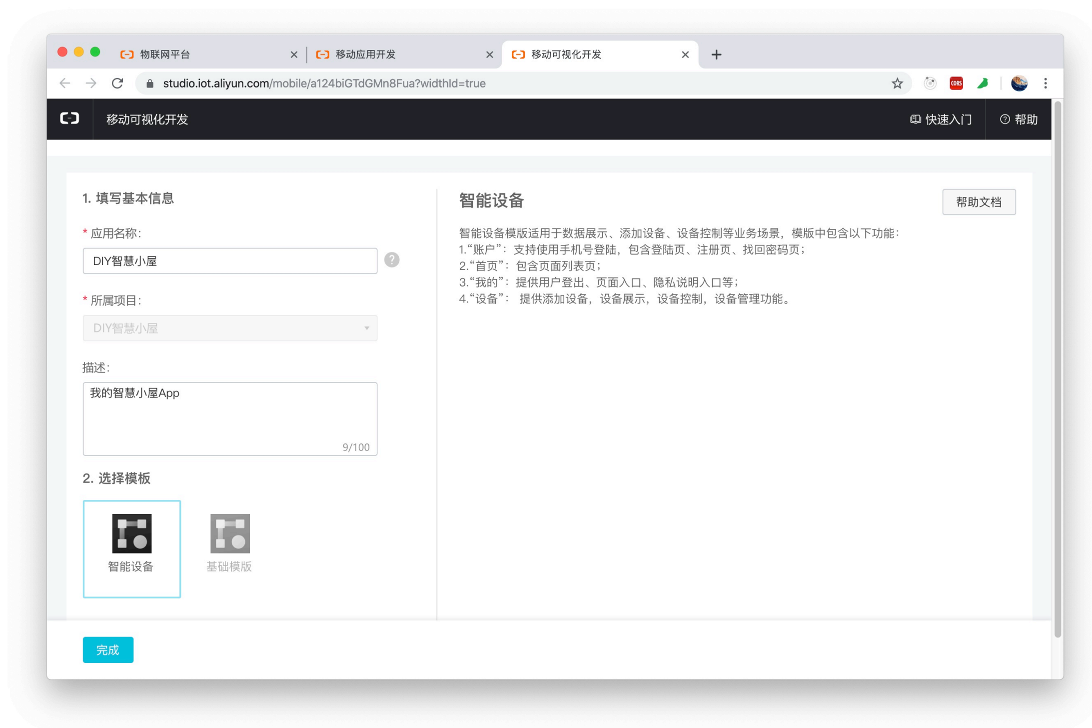
在 IoT Studio 中同样提供了 App 可视化搭建的能力，在智慧小屋中，我们将通过可视化搭建 App 来完成 Web 可视化开发类似的小屋状态查看以及报警查看清除能力。

在项目控制台，选择左侧推荐菜单的移动应用开发，新建可视化应用。



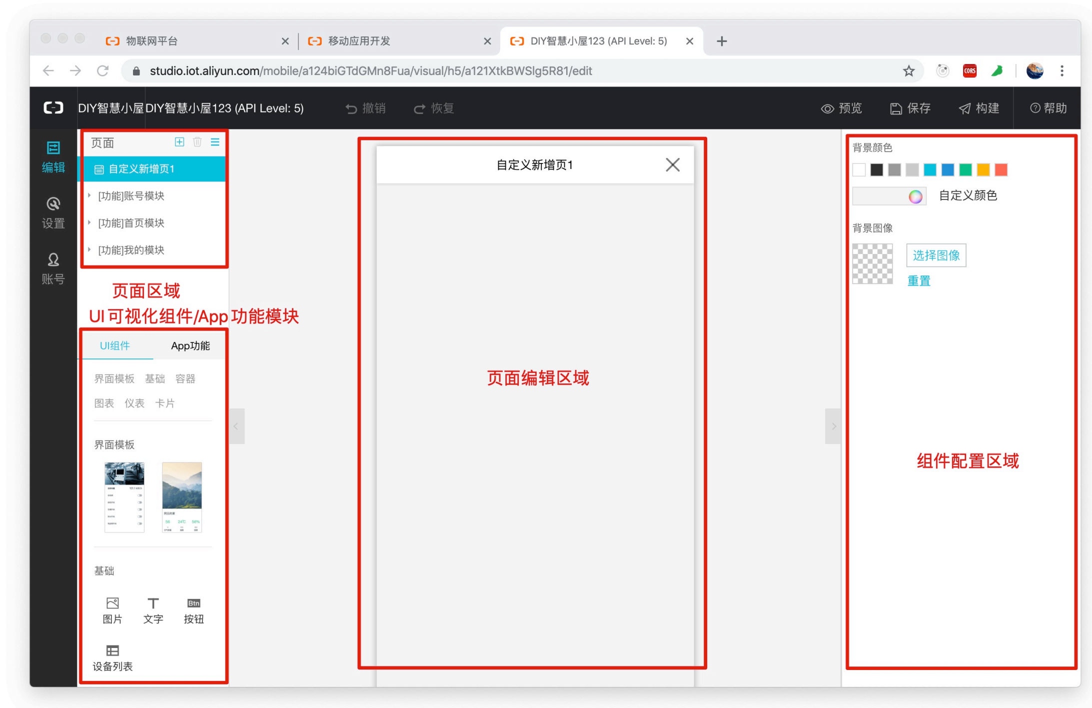
图：移动应用开发

同样，在移动 App 可视化搭建平台也提供了模版，目前提供的有基础模版和智能设备模版，智能设备包含账户、首页、我的（用户中心）和设备管理能力。



图：选择智能设备模版

在进入 IoT Studio 的 App 可视化开发工作台与 Web 可视化搭建的工作台布局类似，在左侧可以对页面进行增删，下方为各类 UI 组件和模版提供的 App 功能模块。



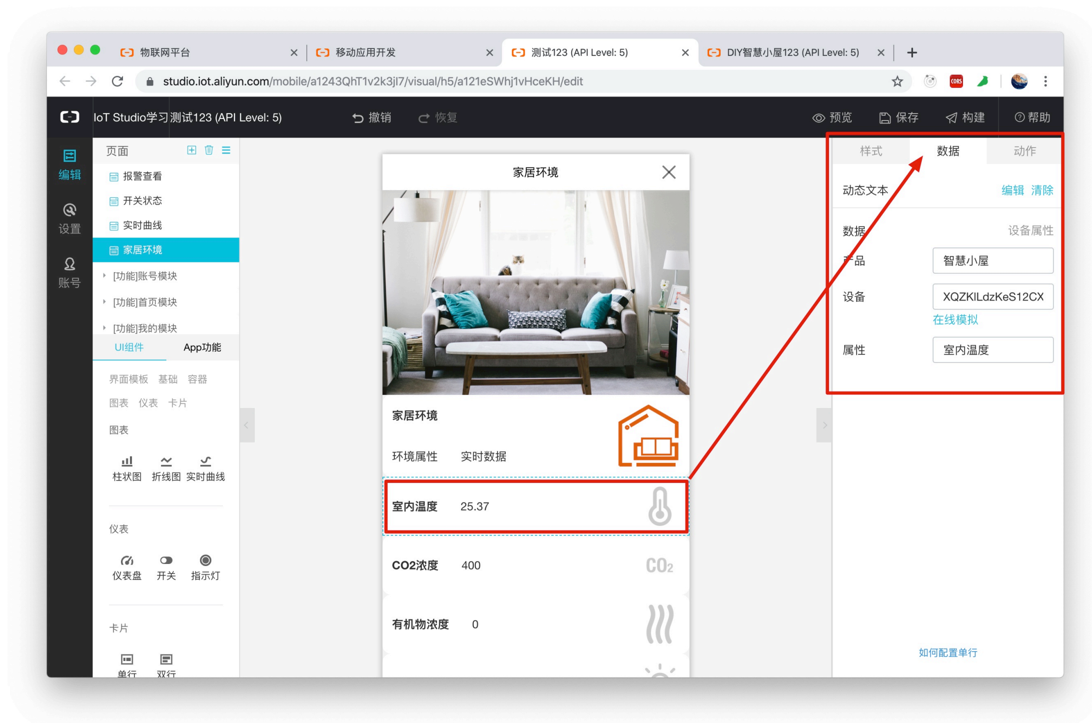
图：App 可视化开发工作台页面结构

App 可视化开发提供的组件有：

- 页面模版：直接集成平台提供的模版。
- 基础组件：提供图片、文字、按钮、设备列表等基础的可视化组件。
- 容器组件：容器组件对页面的结构进行划分，由横向和纵向的容器组成，可以将页面结构划分成为横向和纵向的页面空间。
- 图标组件：图表组件提供了柱状图、折线图、实时曲线等页面常用的图表。
- 仪表组件：仪表组件提供仪表盘、开关、指示灯组件。
- 卡片组件：卡片组件提供单行和双行的页面卡片，可以更美观得展示设备状态和数据。

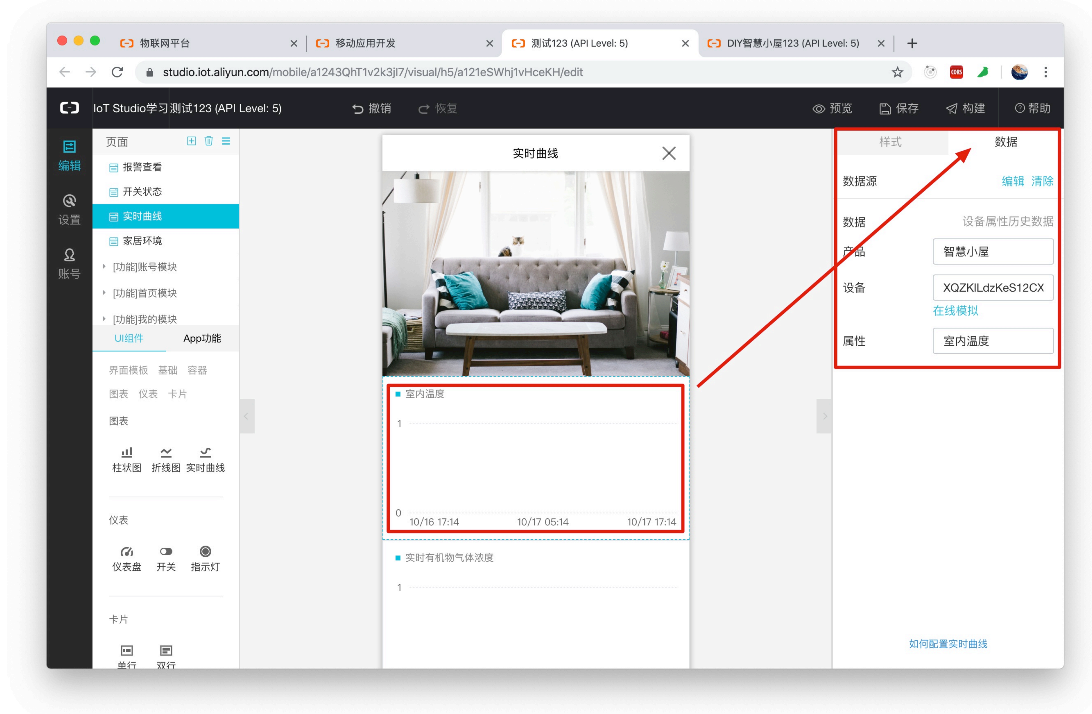
在 App 搭建小屋我们创建四个页面，分别是“报警查看”页面查看燃气有机物浓度并取消蜂鸣器报警；“开关状态”页面查看小屋家电的开关状态；“实时曲线”页面查看室内传感器采集到浓度信息的实时变化曲线；“家居环境”页面也是主页面，展示家居环境主要指标的数据，也是用户进入 App 的主页。

“家居环境”页面采用图片组件和双行以及单行卡片组件，在布局方式采用竖排排列，在单行卡片组件上展示小屋设备上报的各类传感器读数，需要在选中组件后在右侧的数据栏进行数据的配置。



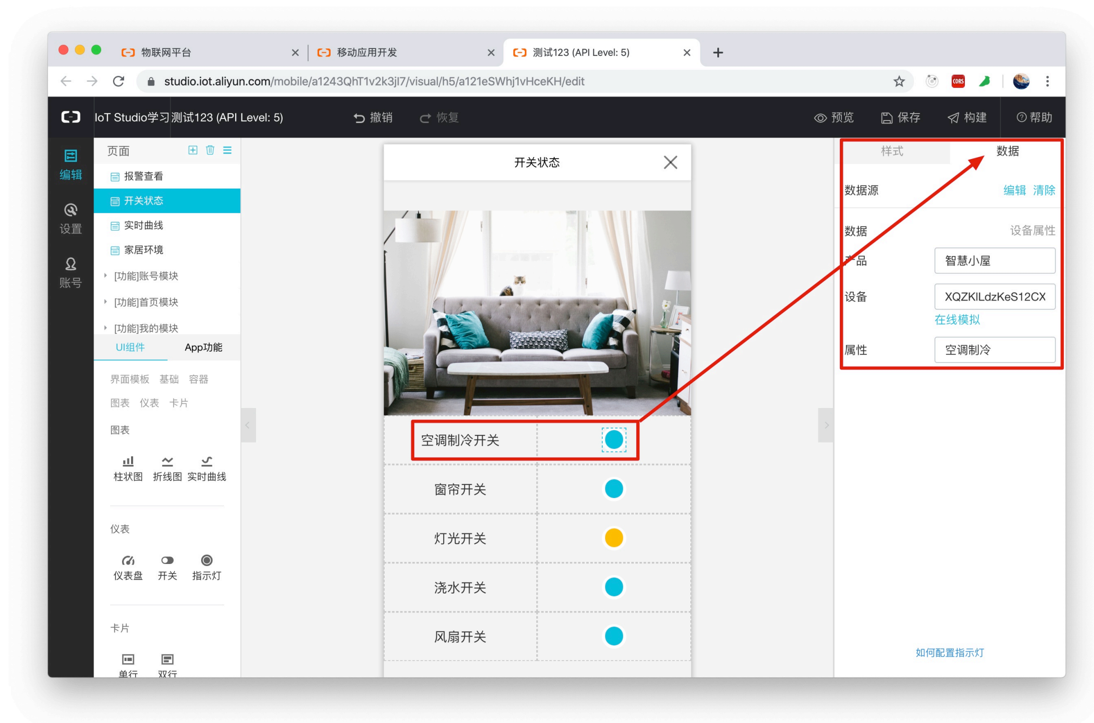
图：家居环境单行卡片的数据配置

“实时曲线”页面展示温度、有机物气体浓度、烟雾传感器读数的实时曲线，使用实时曲线可视化组件和图片组件搭建，在选中可视化组件后在右侧配置区域中的数据绑定智慧小屋的设备信息。



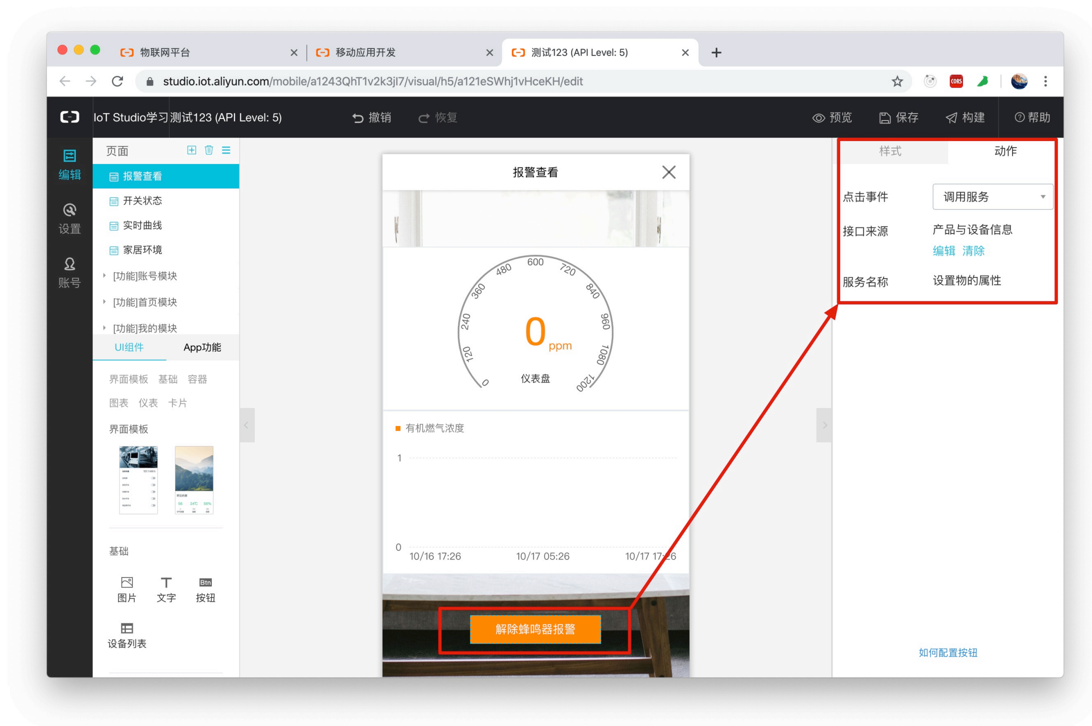
图：曲线图组件绑定设备属性

“开关状态”页面显示智慧小屋各类执行开关的状态，通过容器组件对页面进行划分，使用文字组件和指示灯组件对各类开关状态进行展示，指示灯组件选中后，在右侧的配置区域绑定智慧小屋的开关属性。



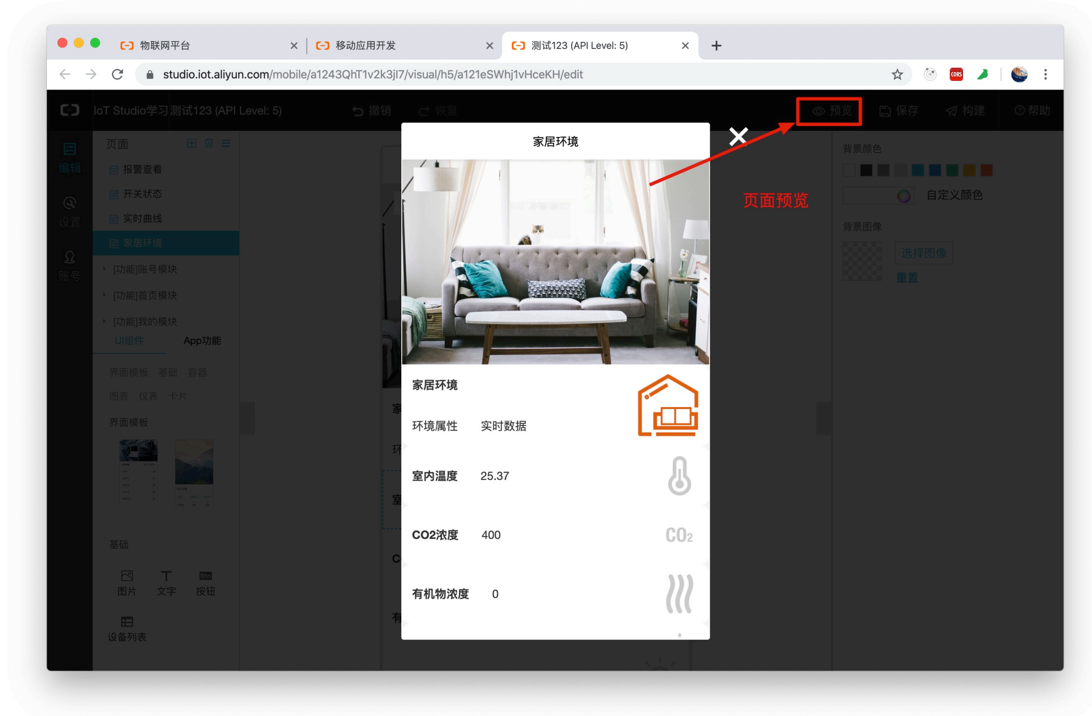
图：开关状态页面指示灯组件数据绑定

“报警查看”页面通过仪表盘显示实时可燃气浓度（有机气体浓度），通过曲线图显示有机物浓度变化曲线，并提供按钮解除蜂鸣器的报警。使用到仪表盘组件、实时曲线组件以及按钮组件，仪表盘和曲线图数据绑定与上面一致，按钮组件选中后，在动作配置中选择调用服务->产品与设备信息->设置物的属性。



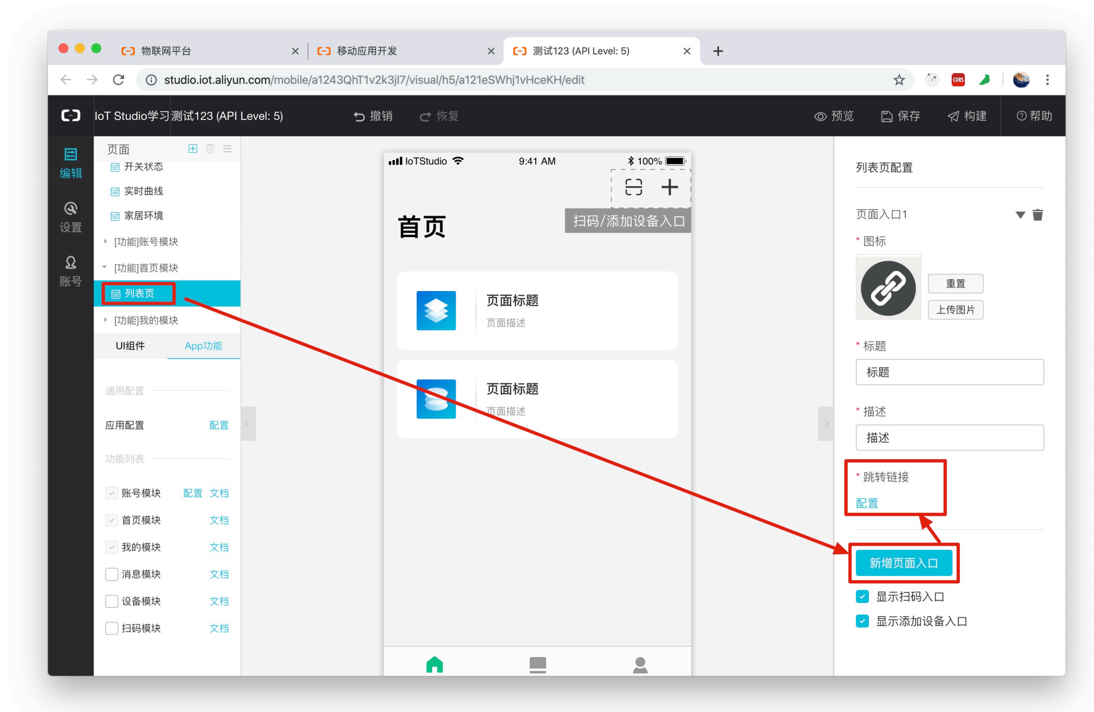
图：配置按钮调用设备服务

在四个页面的编辑中，可以随时点击右上角的预览按钮，对已有的页面进行预览查看，在预览页面上可以实时获取到设备上报的数据内容，并且可以在预览页面直接与设备交互，进行调试。



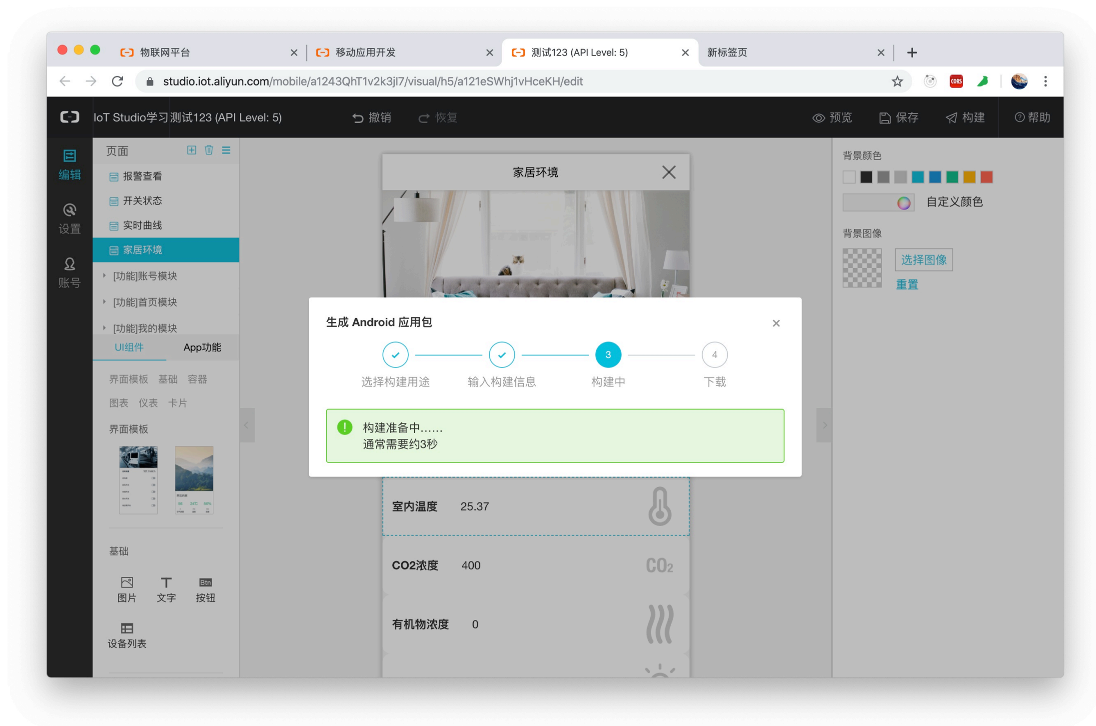
图：页面预览

页面完成添加后，我们需要在 App 的导航页面添加创建的四个页面，在页面栏中选中“首页模块”中的列表页，在右侧点击“新增页面入口”添加页面信息，在上传四个页面中各个页面的图标、配置页面标题和描述后，通过跳转链接将此导航分别关联到前面创建的四个页面。



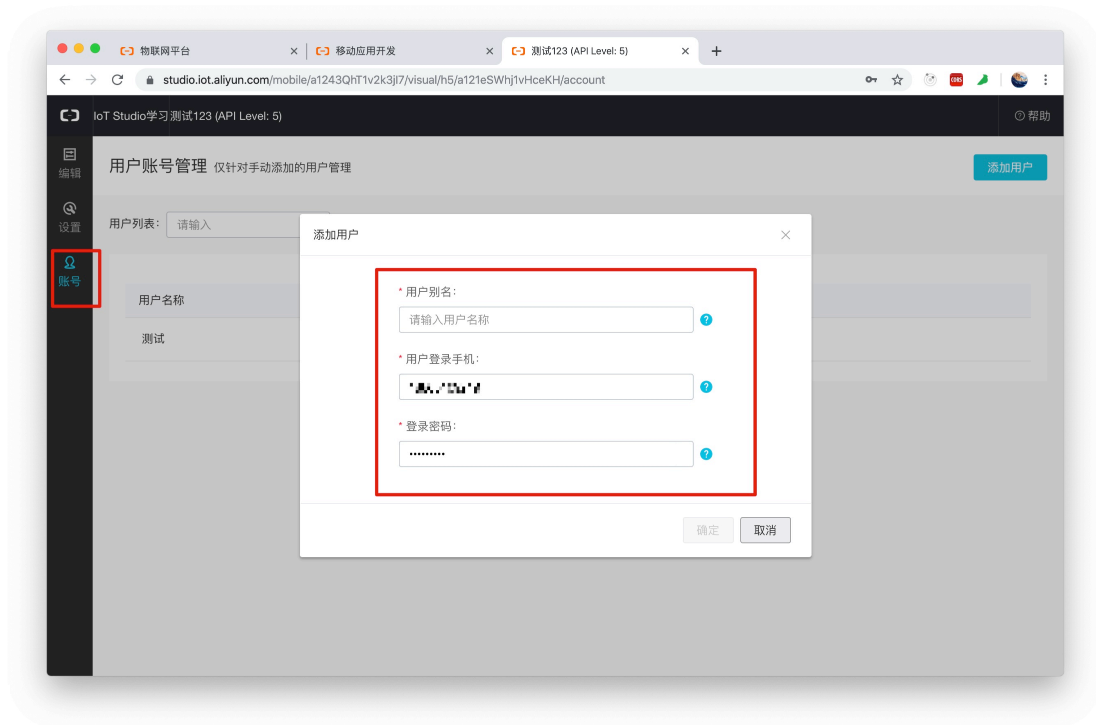
在预览调试结束后，可以对 App 进行构建和下载体验，IoT Studio 提供了一键构建 Android 和 iOS 两个平台应用的能力，Android 可以直接在线构建程序 apk 安装包下载安装，iOS 设备会在线生成程序源码，下载源码后需要使用 xcode ide 软件本地编译构建，具体操作请参见：

https://studio.iot.aliyun.com/studiomobile-doc#mobile-wswg-ios-source-compile-install.html



图：在线构建 Android 版本的 App

App 构建完成后，我们还需要在 App 可视化开发工作台创建 App 的使用用户，用户名和密码信息。点击左侧菜单“账号”选项，在弹出窗口中输入用户昵称、登陆手机和密码，在下载 App 后即可使用此账号登陆，访问各个页面。



图：创建 App 用户账号

至此，我们已经完成 App 的可视化搭建了，大家可以基于上面的介绍 DIY 自己的 App 能力，完善智慧小屋的使用场景。

附件 1、智慧小屋材料清单

本次智慧小屋中使用到的硬件设备清单如下：

序号	名称	数量	购买参考链接	参考事项
1	开关电源	1	https://item.taobao.com/item.htm?spm=a1z09.2.0.0.69332e8djJ8AsW&id=545307800878&_u=j1h4rf77ba56	
2	BME280 环境检测传感器子板	1	https://detail.tmall.com/item.htm?id=599861101484&spm=a1z09.2.0.0.6fff2e8dkPsndV&_u=ifbifqr5ad3	
3	水泵+土壤湿度传感器	1	https://detail.tmall.com/item.htm?id=17619369304&spm=a1z09.2.0.0.69332e8djJ8AsW&_u=j1h4rf774006&skuId=4311412923214	购买自动灌溉 DIY 套餐
4	可燃气体传感器	1	https://item.taobao.com/item.htm?spm=a1z09.2.0.0.69332e8djJ8AsW&id=558794686519&_u=j1h4rf77a680	
5	PM2.5 传感器（使用 CCS811 二氧化碳传感器替代）	1	https://detail.tmall.com/item.htm?id=559012294956&ali_refid=a3_430583_1006:1109983619:N:UvEOWVeLAbmebNLvTwGS3A==:57794b99439d2b04cccdbb3d7bd482da&ali_trackid=1_57794b99439d2b04cccdbb3d7bd482da&spm=a230r.1.14.31	
6	光敏传感器	1	https://detail.tmall.com/item.htm?id=20885572596&ali_refid=a3_430583_1006:1104520036:N:TnKrLEthbIKu6NApBfEhYn2trDRs9/qc:8f2fccb4333de0a35165f29ae2b6e2ad&ali_trackid=1_8f2fccb4333de0a35165f29ae2b6e2ad&spm=a230r.1.14.3&skuId=3946942453971	
7	空调/半导体制冷器	1	https://item.taobao.com/item.htm?spm=a1z09.2.0.0.69332e8djJ8AsW&id=521045583034&_u=j1h4rf770798	建议购买组件，电源可以不买，上面的电源已提供足够的功率
8	换气扇	1	https://item.taobao.com/item.htm?spm=a1z09.2.0.0.69332e8djJ8AsW&id=575260423803&_u=j1h4rf772e41	

9	蜂鸣器	1	https://detail.tmall.com/item.htm?id=575055413166&spm=a1z09.2.0.0.69332e8djJ8AsW&_u=j1h4rf772a9d	可以不购买，系统版上有一个
10	步进电机（窗帘）	1	https://detail.tmall.com/item.htm?id=591695441384&spm=a1z09.2.0.0.69332e8djJ8AsW&_u=j1h4rf77cd9f&skuId=4236401846677	购买电机即可，驱动有驱动板
11	LED 灯	3	https://item.taobao.com/item.htm?spm=2013.1.w4004-14249071599.4.bc5d4df676lyqM&id=524374933434	红色、蓝色、绿色灯各一只（铝基板焊线），功率为 1W 三色灯用于程序控制调色
12	驱动控制开发板	1	https://item.taobao.com/item.htm?spm=a1z10.3-c.w4002-1899881951.64.7feb1c3dzQxXBP&id=606364362084	用于窗帘步进电机、LED 灯驱动、风扇驱动、制冷空调驱动 淘宝店： 晶翔小店
13	Arduino	1	ArduinoUNO 开发板（ATmega328P）和 ArduinoMega 2560 开发板	自己按照型号找合适的即可
14	WiFi 传输模块	0	驱动板已集成	不用单独购买，在驱动板集成

表：DIY 智慧小屋材料清单